RICHARD WILLIAM VALDIVIA

SOLUÇÃO TECNOLÓGICA PARA PRODUÇÃO E ANÁLISE DE UMA REDE DE COLABORAÇÃO A PARTIR DE DADOS DA PLATAFORMA LATTES

Dissertação apresentada à Universidade Federal de São Paulo – Escola Paulista de Medicina, para obtenção do título de Mestre Profissional em Tecnologia, Gestão e Saúde Ocular

São Paulo

2019

RICHARD WILLIAM VALDIVIA

SOLUÇÃO TECNOLÓGICA PARA PRODUÇÃO E ANÁLISE DE UMA REDE DE COLABORAÇÃO A PARTIR DE DADOS DA PLATAFORMA LATTES

Dissertação apresentada à Universidade Federal de São Paulo – Escola Paulista de Medicina, para obtenção do título de Mestre Profissional em Tecnologia, Gestão e Saúde Ocular

Orientadora:
Profa. Dra. Maria Elisabete Salvador Graziosi
Co-orientador:
Prof. Dr. Fábio Luís Falchi de Magalhães

São Paulo

2019

Ficha catalográfica elaborada pela Biblioteca Prof. Antonio Rubino de Azevedo,
Campus São Paulo da Universidade Federal de São Paulo, com os dados fornecidos pelo(a) autor(a).

Valdivia, Richard William

Solução Tecnológica para produção e análise de uma rede de colaboração a partir de dados da Plataforma Lattes – São Paulo, 2019

Dissertação (Mestrado Profissional) - Universidade Federal de São Paulo, Escola Paulista de Medicina. Programa de Pós-Graduação em Mestrado Profissional. Departamento de Oftalmologia. Programa Tecnologia, Gestão e Saúde Ocular.

Título em inglês: Software agent for extraction and quantitative and qualitative analysis of valuated data applied on a Lattes Platform in the health area.

1.Linguagens de Programação, 2. Interface Usuário-Computador, 3.Processamento de Texto, 4.Saúde Ocular, 5.Rede Social, 6.Programas de Pós-Graduação em Saúde, 7.Oftalmologia, 8.Medicina.

UNIVERSIDADE FEDERAL DE SÃO PAULO

MESTRADO PROFISSIONAL EM TECNOLOGIA, GESTÃO E SAÚDE OCULAR

Chefe do Departamento:

Prof. Dr. Mauro Silveira de Queiroz Campos

Coordenador do Mestrado Profissional:

Prof. Dr. José Álvaro Pereira Gomes

RICHARD WILLIAM VALDIVIA

SOLUÇÃO TECNOLÓGICA PARA PRODUÇÃO E ANÁLISE DE UMA REDE DE COLABORAÇÃO A PARTIR DE DADOS DA PLATAFORMA LATTES

Presidente da Banca

Prof. Dra. Maria Elisabete Salvador Grasiosi (Orientadora)

Banca Examinadora

Prof. Dr. Felipe Mancini – Professor Adjunto da Universidade Aberta do Brasil UAB da Universidade Federal de São Paulo

---

Prof. Dr. Flávio Eduardo Hirai – Professor Orientador do Programa de Mestrado Profissional em Tecnologia, Gestão e Saúde Ocular – Universidade Federal de São Paulo

---

Prof. Dr. Marcos Antonio Gaspar – Professor e Docente Permanente do Programa de Pós-graduação em Informática e Gestão do Conhecimento - Universidade Nove de Julho

---

Suplente:

Prof. Dr. Paulo Schor – Professor Orientador do Programa de Mestrado Profissional em Tecnologia, Gestão e Saúde Ocular – Universidade Federal de São Paulo

---

DEDICATÓRIA

"Disse Deus: faça-se a luz, e fez-se a luz. E viu Deus que a luz era boa, e dividiu a luz das trevas. E chamou a luz de dia e as trevas de noite" (Genesis 1, 3, 4, 5).

Para todo fim existe um início e se não fosse aqueles que estiveram comigo no começo dessa jornada, jamais o objetivo seria alcançado com o êxito que o projeto reclamava.

À minha família essa dedicatória segue em especial gratidão, pois além de base de valores são eles quem fazem minha vida e meu trabalho valer a pena. A minha mãe Maria Aparecida Valdivia que me trouxe à essa aventura que muitos chamam de "vida". Aos meus irmãos Maurício e Humberto que são o centro de meus valores morais e intelectuais. De uma forma ou de outra sempre acreditaram em mim, mesmo quando eu já acreditava que não podia faze-lo. Ao meu filho e pequeno *Jedi,* Gustavo Fausto Valdivia que se apresenta como uma pérola de valor inestimável nesse mundo e que me acompanha mesmo quando no caminho escuro, apenas para se mostrar presente.

Muito a dedicar e agradecer a Marcia Suguimoto, que me sempre me incentivou soube compreender, repreender e colocar nos eixos tudo o que realmente importa. Dona de superlativos e adjetivos dos mais elevados esteve sempre presente durante a produção desse trabalho, muitas vezes sendo a força motriz e ponto de apoio não me deixando esmorecer diante das dificuldades. Não posso deixar de mencionar também a Miwa Suguimoto que com pequenos gestos foi capaz de auxiliar sem nunca questionar meus objetivos, preparando e ajudando em minhas necessidades imediatas sem nunca pedir nada em troca.

AGRADECIMENTOS

Agradeço primeiramente a Deus, essa força divina de forma indefinível que dentro de meu conjunto de crenças guia meu caminho. Os nomes são muitos, entretanto, por motivos de espaço físico nesse trabalho mencionarei apenas alguns poucos que para mim representam marcos e que tiveram papel fundamental no meu desenvolvimento pessoal, mas acreditem: todos que me são importantes estão imortalizados dentro de minha alma, pois o que vale nessa vida é o sentimento, muito pouco importando prova material.

Agradeço a muitas pessoas que passaram em minha vida que apesar do passado distante ainda hoje se fazem lembrar, pois muito contribuíram para o desenvolvimento do meu caráter e minhas propriedades intelectuais. Parafraseando Newton: "*se eu vi mais longe, foi por estar sobre ombros de gigantes*" e por estar sobre "esses" gigantes é que culmina esse momento: primeiro a Profa. Zuleika (quem me ensinou as primeiras letras), o Prof. Escudeiro de matemática na 5º. série do ginásio que o tempo deixou a lembrança de uma pessoa que muito me incentivou, ao Prof. Carlos (SENAC) quem acreditou em meu potencial e me incentivou mais do que eu até mereceria e a quem nunca vou esquecer suas aulas de cálculo, à amiga Alice Akemi Yamazaki (que provavelmente não tem mais esse sobrenome) mas me ajudou a forjar meu caráter e gosto pelos estudos e finalmente ao Prof. Dr. Marcelo Paiva – UNIFESP – quem me mostrou potencial que eu não sabia que tinha.

Hoje, os rostos atuais mudaram, mas o sentimento é o mesmo. Agradeço minha orientadora Profa. Dra. Maria Elisabete Salvador Graziosi e ao meu querido co-orientador Fábio Luis Falchi de Magalhães que juntos souberam coordenar meu trabalho, me ajudaram a organizar as ideias, contribuíram com o aprendizado de forma consistente. Mas agradeço principalmente por estarem presentes no momento certo e quando necessário durante a realização desse projeto. Sou afortunado por tê-los por professores.

Agradeço aos amigos Fábio Santos que com sua conversa demorada sempre me apoiou, ao Ederson Luiz Silva, este que de forma indireta e por meios próprios me incentivou nos meus trabalhos e ao amigo Pedro Robson Leão que representa pra mim uma figura de hombridade e garra no que faz independente do horário, esforço necessário e objetivo proposto.

Estes, são todos alicerces que me ajudaram a construir não ainda um castelo, mas as paredes que dão sustentação e soberania ao meu maior projeto: a minha vida e tudo que nela opera.

À Gustavo Fausto Valdivia, meu filho:

"*Você é luz de meu caminho,*

o sentido de minha vida

e meu mais belo sonho realizado"

ABREVIATURAS

ARS	Análise de Redes Sociais
CNPq	Conselho Nacional de Pesquisa
CAPES	Coordenação de Aperfeiçoamento de Pessoal de Nível Superior
CSS	Cascading Style Sheets
DOM	Document Object Model
DOM4J	Documento Object Model for Java®
EPM	Escola Paulista de Medicina
ES	Engenharia de Software
HTML	Hypertext Markup Language
I/O	Input / Output (Entrada e Saída)
IoT	Internet Of Things (Internet das Coisas)
JRE	Java® Runtime Environment (Ambiente de execução Java®)
JVM	Java® Virtual Machine
MA	Mestrado Acadêmico
MP	Mestrado Profissional
OO	Orientação a Objetos
POO	Programação Orientada a Objetos
PPG	Programa de Pós-Graduação
SAX	Simple Api for XML
UNIFESP	Universidade Federal de São Paulo
W3C	World Wide Web Consortium
WWW	Word Wide Web

LISTA DE FIGURAS

LISTA DE TABELAS

SUMÁRIO

RESUMO

Com a evolução de técnicas computacionais é imperioso que ferramentas automatizadas sejam criadas para promover valor ao grande conteúdo informacional encontrado na Web utilizando linguagens de programação. O desenvolvimento de instrumental tecnológico a fim de responder às questões de tomadas de decisão é hoje uma realidade. O presente estudo analisou as atuais técnicas e processos utilizados para exploração e leitura de dados distribuídos na internet e propôs uma ferramenta automatizada para gerar informação produtiva e relevante da Plataforma Lattes através da análise de processamento de texto. A ferramenta recuperou informações estruturadas para uso em Programas de Pós-Graduação em Saúde na área de Medicina – Oftalmologia e Saúde Ocular - e gerou gráficos de redes de colaboração de pesquisadores Stricto Sensu como estudo de caso obtido por referencial teórico.

Palavras chave: Linguagens de Programação, Interface Usuário-Computador, Processamento de Texto, Saúde Ocular, Rede Social, Programas de Pós-Graduação em Saúde, Oftalmologia, Medicina.

ABSTRACT

With the evolution of computational techniques it is imperative that automated tools be created to promote value to the great informational content found on the Web using programming languages. The development of technological instruments in order to answer the questions of decision-making is now a reality. The present study analyzed the current techniques and processes used for the exploration and reading of data distributed on the Internet and proposed an automated tool to generate productive and relevant information of the Lattes Platform through the analysis of word processing. The tool retrieved structured information for use in Post-Graduation Programs in Health in the area of Medicine - Ophthalmology and Ocular Health - and generated graphs of collaboration networks of Stricto Sensu researchers as a case study obtained by theoretical reference.

Keywords: Programming Languages, User-Computer Interface, Word Processing, Eye Health, Social Networking, Health Postgraduate Programs, Ophthalmology, Medicine.

1 INTRODUÇÃO

1.1 Contextualização

Com a crescente necessidade dos Programas de Pós-Graduação atender às normativas da Coordenação de Aperfeiçoamento de Pessoal de Nível Superior (CAPES) a fim de manter a excelência dos cursos em termos de notas necessárias para atividade dos Programa, a Plataforma Lattes se apresenta como um repositório útil e rico de informações que interessa às universidades a fim de se obter conhecimento sobre o que está sendo realizado no âmbito acadêmico ao mesmo tempo em que acompanha o desempenho da produção cientifica dos principais atores e produtores de artigos dessas instituições Capes (2018) (1)

Dessa forma, com o atual estado da arte da Tecnologia da Informação, fica claro que uso de técnicas computacionais para analisar volumes de informações cientométricas, torna o moroso trabalho de coleta e organização dos dados em algo de fácil manipulação após o processamento, tornando assim possível a quantificação e avaliação da qualidade das produções cientificas produzidas.

Percebeu-se então uma lacuna de pesquisa que trata de forma específica da criação de instrumental tecnológico, no que tange ao desenvolvimento de ferramentas automatizadas, para gerar informação útil com o conteúdo de dados disponíveis na World Wide Web (WWW) através da agregação, compilação e processamento de informações em massa para que os resultados dos dados se transformem em informação relevante para Programas de Pós-Graduação em universidades brasileiras.

Para o esforço proposto foi verificado duas tendências que forma seu núcleo principal. O primeiro é validar e dar o devido reconhecimento aos pesquisadores de alto desempenho em universidades brasileiras, recuperando suas informações de produções científicas, permitindo analisar de maneira quantitativa e qualitativa sua evolução ao longo do tempo, sua colaboração em pares e orientação de alunos.

Uma segunda perspectiva é poder analisar, o produto extraído e compilado da internet, de informações criadas dentro das instituições do ponto de vista quantitativo e qualitativo. É através da agregação das informações e sumarização dos dados de um grande número de pesquisadores que resultados são obtidos, fornecendo dessa forma uma maneira fácil e prática para que Coordenadores de Programas de Pós-Graduação possam, a partir dos dados sumarizados, criar políticas e normativas que atendam aos preceitos de avaliação da CAPES.

Assim, este se torna um desafio importante a realização da coleta de informações e disponibilização organizada, intuitiva e automatizada, a fim de que coordenadores de Programas de Pós-Graduação possam visualizar as informações e realizar a tomada de decisão de forma assertiva.

Observa-se que a informação distribuída pela World Wide Web (WWW) ou comumente denominada de Internet vem crescendo a cada dia e, portanto, a necessidade de se criar mecanismos para exploração de informações com uma busca eficiente centrada na necessidade do usuário é uma demanda exigida.

Anteriormente, nos primórdios da Internet, a pesquisa de dados era difícil e trazia pouca informação útil. Atualmente, a busca convencional de dados na Internet se baseia no modelo léxico do conjunto de palavras chave utilizado pelo usuário, aliado a robustos sistemas de *query/search* provido por sistemas computacionais como o Google® e outras ferramentas de extração de informações (Bonifácio 2002 (2)), sendo assim importante que novas ferramentas sejam criadas para atender à crescente demanda de dados.

Entretanto esse modelo de armazenamento, apesar de ser viável para os usuários, não representa as melhores práticas de recuperação e refinamento do dado que é resultante dessa busca quanto esse cenário é apresentado juntamente com o advento do Big Data, ou grandes quantidades de informações que precisam ser processadas. Percebeu-se que na WWW atual não existe critério no armazenamento das informações ou uma padronização o que torna o processamento e recuperação de dados por meios automatizados pouco produtivo e com um nível de precisão baixo (Lima 2005) (3)).

A solução para esse problema seria então a recuperação da informação através de metadados (Berners-Lee 2001 (4)) para serem processados por máquinas. Ainda segundo Berners-Lee (2001) (4) o desafio é fornecer informação adicional para que sistemas automatizados sejam capazes de realizar inferências e obter melhores resultados e linguagens para prover os mecanismos necessários para essas consultas.

Nesse contexto, o repositório como a Plataforma Lattes torna-se um excelente candidato para pesquisa e análise de informações, uma vez que sua estrutura está organizada e é possível extrair os dados em formato de metadados eXtensible Markup Language (XML) e possui informação pertinente a ser processada, gerando valor no resultado das consultas através de uma ferramenta computacional capaz de ler esse tipo de informação estruturada.

Pesquisas apontam que existe um consenso acadêmico que o atual cenário irá mudar para um modelo baseado em conhecimento conhecido como Web Semântica e com técnicas aplicadas ao Big Data, Linked Data, Linked Open Data, permitindo a inferência na busca de informação (Breitman 2005 (5)). O que conhecemos como Internet atualmente poderá ser reescrita para que haja interoperabilidade e troca de informações entre sistemas e assim realizar integração entre os dados e o seu significado tornando dessa forma, os dados uteis para tomadas de decisão (Berners-Lee (2001) (4)). Entretanto o saber computacional atual ainda está muito pautado em busca estruturada em formatos como Json e XML.

O Conselho Nacional de Pesquisa (CNPq) tem feito um enérgico trabalho ao criar um repositório de currículos de pesquisadores permitindo uma visualização temporal da informação. O Currículo Lattes possui em seu repositório, atividades atuais e pregressas do orientador, dessa forma podendo realizar análises sobre diversas perspectivas da pesquisa brasileira. Com a adoção de um padrão na organização das informações o Currículo Lattes se torna um ótimo candidato de análise e conteúdo utilizando ferramentas automatizadas.

A análise cientométrica, ou seja, aquela que possui dados sobre a perspectiva cientifica da instituição, possui um papel importante nas academias e é importante que existam ferramentas apropriadas para a coleta, catalogação e tratamento da informação da base de dados de pesquisadores. Conforme Ribeiro (2013) (6), esses dados são a referência para o fomento de bolsas para os cursos de Pós-Graduação. Também se observa que o processo de monitoramento da atividade cientifica permite verificar e validar o estágio de desenvolvimento tecnológico do país e obter um recorte das pesquisas realizadas em seus diversos períodos (Machado 2005 (7)).

Portanto uma ferramenta para extração dos dados do Currículo Lattes e posterior análise quantitativa e qualitativa é de interesse das universidades brasileiras uma vez que permite monitoramento de informações de necessidade de Programas de Pós-Graduação.

Como resultado dessa pesquisa será proposto ao final uma peça de sofware que será capaz de ler o conteúdo dos dados da Plataforma Lattes ao realizar o acoplamento e interoperabilidade com outras ferramentas computacionais e gerar informação útil sobre os pesquisadores além de exportar dados para outros sistemas para alimentar novas pesquisas.

Esse trabalho propõe a realização de implementação de uma ferramenta tecnológica que de forma automatizada, processa e gera resultados a partir da extração informações de

pesquisadores na Plataforma Lattes. O escopo de dados se restringe às informações da área da saúde, porém, um estudo de caso deve atender a análise de informações de qualquer área.

Para que o objetivo fosse atingido estabeleceu-se uma ordem de organização de capítulos onde é apresentado inicialmente o referencial teórico necessário para atingir a argumentação e que mais tarde é utilizado na fase de descrição da metodologia, assim como o referencial foi importante para apresentação e validação dos resultados e a conclusão final do trabalho.

Importante lembrar que em cada capítulo o autor apresenta apenas os tópicos que são relevantes para essa pesquisa, porquanto alguns desses capítulos seriam demasiados longos para que fossem explorados na sua totalidade e não trariam benefícios para as conclusões finais. Um exemplo é o Capítulo 2.4 Arquiteturas para soluções tecnológicas de software onde é apresentado os Processos de Desenvolvimento de Software. Foram apontados somente os casos clássicos e suficientes para que servisse de embasamento teórico para essa pesquisa.

O referencial teórico forma o arcabouço de conhecimento necessário para organização e apresentação dos resultados foi dividido em três partes principais.

Primeiro foi realizado o levantamento de artigos relacionados com aspectos das Redes de Colaboração que ao final dessa pesquisa é utilizada como estudo de caso para apresentação dos resultados práticos.

Depois foi dissertado sobre o cenário da Pós-Graduação no Brasil e suas particularidades e divisões em níveis e os tópicos importantes sobre a evolução e avaliação realizada pela CAPES para continuidade de fomento nos Programas de pesquisa ofertados pelas universidades brasileiras.

Ao final e como parte com maior conteúdo é descrito as técnicas e ferramentas relacionadas com a Engenharia de Software e aspectos técnicos utilizados para construir uma ferramenta de software com capacidade para organizar e extrair conhecimento de informação textual da Plataforma Lattes e geração de conhecimento útil para uma análise mais aprofundada do ponto de vista quantitativa e qualitativa.

2 REVISÃO DA LITERATURA

2.1 Pós-Graduação, pesquisa científica e Redes de Colaboração

As redes de colaboração podem ser definidas como a correlação de pares de indivíduos ou de grupos de tal forma que troquem informações e se auxiliem mutuamente no processo de interação entre os elementos que compõe o domínio de integrantes.

Na pesquisa cientifica, as redes de colaboração são essenciais para a construção de novos conhecimentos baseados na relação e troca de informações entre pesquisadores e objetivos comuns, como forma para se alcançar metas relacionadas ao desenvolvimento tecnológico de determinada área.

2.1.1 Redes de colaboração em ambientes organizacionais e institucionais

É notório que cada vez mais a organizações tem se concentrado em organizar estratégias com capacidades para melhorar seus processos e consequentemente serem capazes de realizar tomadas de decisão de forma assertiva com altas taxas de precisão, tornando essas cada vez mais competitivas e representativas no segmento em que estas propões seus resultados (Francisco (2018) (8)).

Assim, Programas de Pós-Graduação também podem utilizar tais técnicas para monitorar seu desempenho e transformar seus resultados através do envolvimento de pesquisa e pesquisadores e é sabido que ambos (pesquisadores e instituições) utilizem redes de colaboração para observar potenciais em tais ambientes.

No âmbito das universidades as redes de colaboração ou esforços colaborativos entre pesquisadores são importantes pois apresentam de forma gráfica e de fácil entendimento as responsabilidades e desafios que os grupos de pesquisa enfrentam ao realizarem interatividade entre si, seja através de grupos de estudo ou mesmo de indivíduos, ao realizarem seus trabalhos dando mérito a todos do grupo com os esforços compartilhados entre os pares (Balancieri (2005) (9)).

Em uma perspectiva mais ampla as redes de colaboração denotam também, além dos atores envolvidos nas pesquisas e sua relação entre membros dos grupos de pesquisadores, a união de competências que resultam na criação de conhecimento capaz de gerar novos procedimentos de inovação sendo esses os resultados das inovações tecnológicas observáveis nas áreas do conhecimento. A inovação é produto das interações dos pesquisadores juntamente com os esforços em metas correlatas.

Observa-se então que há um enorme ganho ao utilizar essa técnica de análise pois dela resulta num aumento significativo de abordagens sobre um determinado tema. Abre espaço para a concepção cruzada de novas formas de se ver um problema científico quando os esforços partem de grupos diferentes visões. A maneira que se trabalha para alcançar essa meta é quando então os grupos que trabalham em uma área, buscam referências em outros grupos com o mesmo objetivo, mas acrescentam direções não percebidas ainda. Isso é possível pois novos conhecimentos são apresentados pelos outros atores mas agora com um novo olhar.

Segundo Maia (2011) (10) o que vemos na atualidade é um elevado número de publicações que são impulsionados pela capacidade de intercambio que os pesquisadores realizam entre si. O fator econômico é um elemento importante, pois ao posicionar grupos de indivíduos para trabalho em conjunto há uma economia de tempo e dinheiro para que os trabalhos atinjam seus fins. A pertinência dessa economia encontra nas agências financiadoras de pesquisas vantagens importantes.

2.1.2 Análise de Redes Sociais

Uma metodologia poderosa para ser aplicada às redes de colaboração é a Análise de Redes Sociais (ARS). Essa abordagem teve sua concepção inicial na área de computação, com estudos relacionados aos circuitos eletromagnéticos e inicialmente denominada como abordagem por Análise de Redes (AR) que tem como premissa estudar relações entre pares de objetos com natureza semelhante (Balancieri (2004) (11)).

Atualmente a ARS tem se demonstrado um método importante para se entender as relações de problemas complexos nas relações de entidades humanas e sociais. Esse ajustamento entre a Análise de Redes e a Análise de Redes Sociais permitiu criar uma ferramenta capaz de apresentar soluções para diversas questões encontradas na sociedade de forma que apresenta como elo de ligação para grupos e indivíduos em um determinado contexto. A ARS é também uma forma de poder organizar conceitos subjetivos que são provenientes das áreas sociais em sistemas estruturados o que permite realizar correlações em uma análise qualitativa e fornecer subsídios fortes sobre as relações e seu comportamento observado.

Segundo Balancieri (2004) (11), apesar da representação que permite utilizar as técnicas de ARS e do enorme crescimento de estudos correlatos, os pressupostos teóricos ainda carecem

de bases teóricas consistentes sendo que estudos empíricos são os que são mais largamente apresentados.

A abordagem de análise também possui em seu tronco principal raízes ligadas a sociometria e teoria dos grafos com uma visão analítica de natureza matemática. Com uma metáfora que organiza e estrutura as interpelações entre indivíduos, é possível visualizar valores diferentes em um grupo que aparentemente é homogêneo.

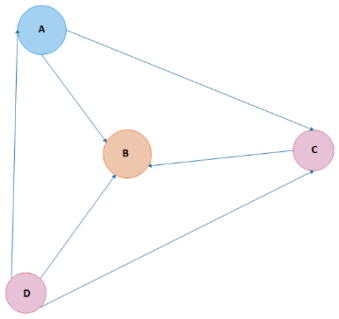


Figura 1: Rede de colaboração baseado na teoria dos grafos. Fonte: próprio autor. São Paulo, SP. 2019

A Figura 1 demonstra mostra a relação entre indivíduos utilizando-se a representação de grafo de rede onde cada ponto ou vértice representa um indivíduo e a linha que conecta cada elemento denota a inter-relação entre dois indivíduos (Cevantes 2015 (12)). A representação do círculo ou nó é denominada como vértice da rede e as linhas como arestas que ligam os vértices.

Historicamente o que é chamado de ARS atualmente, tem como primeiro estudo o texto publicado por Leonhard Euler 1790 no qual ele discute um problema chamado de “Sete pontes de Konisgsberg” (Recuero 2017 (13)).

Trata-se de um jogo onde uma cidade de chamada Konisgsberg está marcada por um total de sete pontes. O objetivo é conseguir atravessar as sete pontes sem caminhar por uma rua mais de uma vez. Nesse jogo cada ponte é representada por um vértice e as ruas que interligam as pontes são as arestas. O interessante desse desafio é que Euler demonstrou que o jogo não é passível de solução baseando-se em conceitos matemáticos e modelados por ele. Esse foi o início da teoria dos grafos largamente utilizada nas Ciências da Computação para análise de tráfego de rede.

Essa visão de relacionamento em rede apresentada por Euler (Cevantes 2015 (12)) é uma premissa importante e adaptada à AR e ARS para dar sustentação ao discurso da análise

de interrelações. De fato, essa visão sistêmica de relacionamento é consistente e permite aferir dados que podem ser subjetivos demais não são organizados de forma estruturada.

2.1.3 Histórico das Redes de Colaboração

Conforme verificado por Balancieri (2004) (11), no final da década de 1950, Smith (1958) (14) foi o pesquisador pioneiro que propôs uma análise em que a colaboração entre grupos de pesquisadores pudesse ser avaliada a fim de se obter informações sobre o crescimento dos resultados das pesquisas. Ademais como essa interrelação de indivíduos caracterizava uma pesquisa científica com maior qualidade (Smith 1958 (14)).

Para Smith (1958) (14) a colaboração entre pares tinha como principal capacidade a transformação do conhecimento em textos e publicações de forma que a elaboração desses resultados seria apresentada de forma mais refinada e com maior garantia no tocante a eficácia do produto ou trabalho produzido.

Durante a década de 1960, vários esforços foram realizados no sentido de observar as relações entre pesquisa e pesquisadores para se entender como as pessoas e organizações estavam se orientando para alcançar os objetivos científicos (Recuero 2017 (13)).

As investigações permitiram que alguns teóricos compreendessem a existência dos chamados “colégios invisíveis” como apresenta Recuero (2017) (13). Essa denominação foi dada ao se perceber que a relação entre os pesquisadores da época se dava em relações informais e não constituíam laços institucionais. As pessoas eram conhecidas umas das outras limitando-se a troca de conhecimento a grupos específicos (Balancieri 2004 (11)) sem a academia como ponto de encontro.

Fica claro que os questionamentos de Smith (1958) (14) estava diretamente relacionado aos resultados que as relações de colaboração alcançavam. Em outras palavras, um grupo coeso de cientistas permite atingir objetivos comuns desses grupos além de ganhos indiretos como uso racional de recursos, economia de tempo o que é atualmente uma premissa almejada pelas agências de pesquisa (Maia 2011 (10)).

A dinâmica de relações modificou-se muito durante a década de 1970. Nesse período a busca principal dessa área estava direcionada em encontrar quais eram as áreas que havia mais cooperação entre pares. A bibliometria foi uma ferramenta fundamental durante esse período. Por esses estudos foi observado um dado importante relacionado à algumas áreas da referida época onde as “ciências básicas e naturais” representavam os cenários onde se encontravam os

maiores núcleos de cooperação e que as “ciências aplicadas e sociais” não alcançavam esse nível de comunicação (Balancieri 2004 (11)).

Balancieri (2004) (11) ensina que nesse período havia fortes convicções que a pesquisa cientifica por coautoria era um objetivo a ser alcançado, pois diversos estudos corroboravam nesse sentido pois foram apresentados resultados onde havia maiores citações de artigos em publicações quando esses artigos vinham calçados em maiores coautorias e menos citações quando os artigos eram publicados com apenas um único autor .

Os anos seguintes demonstraram que a colaborações cientificas deveriam transpor os muros nacionais e, portanto, as coautorias internacionais eram incentivadas. Os estudos relacionados com a temática de redes de colaboração demonstraram que são ótimos os resultados quando existem grupos coesos, integrados e com objetivos afins (Balancieri 2005 (9)).

A sistematização de métodos de sociometria coletadas levaram a seguinte compreensão: um mundo mais integrado é melhor para o crescimento de uma sociedade e, portanto, é pertinente que nos dias atuais, as formas de interação relacionadas com às tecnologias emergentes sejam um instrumento de constante avaliação (Balancieri 2005 (9)).

Assim conforme Oliveira (2017) (15), os desdobramentos da pesquisa dessa temática podem confirmar que as tendências colaborativas vêm aumentando a cada dia pois o ganho é muito compensador para os indivíduos/pesquisadores participantes, para as agências de fomento ao distribuir verbas de pesquisa para grupos de alta performance e por fim toda a sociedade.

2.2 Pós-graduação no Brasil

A Coordenação de Aperfeiçoamento de Pessoal de Nível Superior tem realizado um grande esforço para a manutenção de excelência no quadro de Programas de Pós-Graduação Stricto Sensu através de avaliações desses programas que tem como objetivo apoiar a expansão e qualidade de ensino no Brasil.

Segundo Magalhães (2018) (16), através do amparo aos pesquisadores e docentes a CAPES tem como um de seus objetivos viabilizar a construção de mudanças no que tange os alicerces de avanço no conhecimento e premissas de desenvolvimento da sociedade, articulando meios que promovam os níveis de Mestrado, Mestrado Profissional e Doutorado em resultados de pesquisas de excelência.

Essa observação é notada por Reynault (2014) (17) quando expõe que há um movimento importante da atualidade em que estabelece uma ligação entre o indivíduo e a coletividade num sentido amplo de transformação do modo de pensar e que é caracterizado pela ascendência de novas metodologias e tecnologias de pesquisa acadêmica.

Dessa forma e para atender aos anseios da sociedade que é a grande beneficiada pelos resultados acadêmicos, a CAPES delineia as premissas dos níveis acadêmicos a fim de formar docentes e pesquisadores nos níveis acadêmicos a seguir.

2.2.1 Níveis acadêmicos Stricto Sensu

No nível de Mestrado espera-se que o aluno possa contribuir em avanços do conhecimento, entretanto seu objetivo maior é o de utilizar conceitos e instrumentos metodológicos. Sua principal missão é a de permitir a progressão do conhecimento cientifico (Raynault (2014) (17)).

O Doutorado ainda segundo Raynault (2014) (17) é a linha basilar da produção do saber cientifico e sua difusão. É no Doutorado que a contribuição cientifica se torna obrigatória levando a comunidade a uma nova perspectiva do conhecimento e sua transformação.

No caso do Mestrado Profissional, Magalhães 2018 (16) apresenta que este aparece no cenário acadêmico brasileiro em 1995 como forma a atender determinadas demandas não necessariamente acadêmica, mas com uma visão de prática aplicada, sendo o resultado das pesquisas desse nível de pesquisa absorvido rapidamente pela sociedade. Com característica

tecnicista, o Mestrado Profissional produz resultados práticos a partir de estudos voltados ao desempenho, mas agora com alto nível de qualificação.

Interessante observar também que o Mestrado Profissional tem tido um crescimento relevante e a CAPES tem dado muita atenção para esse campo. De forma prática Magalhães (2018) (16) apresenta em seu trabalho de análise quantitativa, resultados obtidos sobre a evolução e crescimento cientifico no Brasil nos últimos anos com dados coletados compreendendo o período de 1960-2017 em um extrato apenas na área de Tecnologia da Informação apresentado na Figura 2.

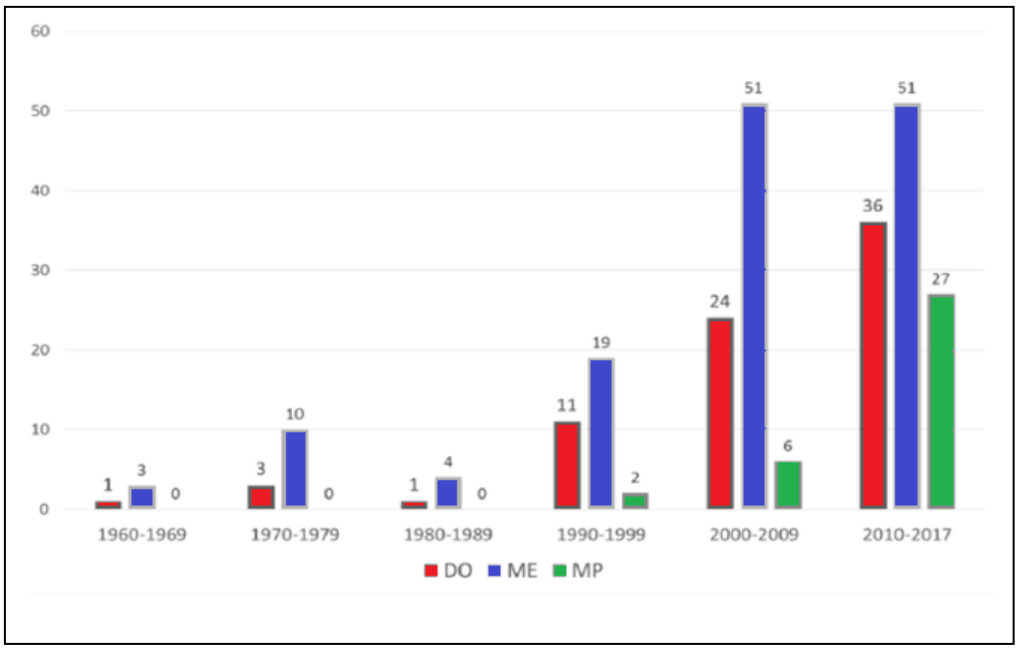


Figura 2: Evolução do Stricto Sensu em TI: Fonte: Magalhães (2028) (16)

Segundo Magalhães (2018) (16), o crescimento dos Programas tem uma evolução da ordem de 200% nos níveis mais tradicionais sendo o Mestrado Acadêmico com 268%, o Doutorado com 218% e o Mestrado Profissional que começou tímido mas entre 2010-2017 um aumento de 450% de oferta de novos cursos.

Essa estratificação em dados somente para a área da Tecnologia da Informação é importante para demonstrar o crescimento significativo do Mestrado Profissional já que essa área demanda muito conhecimento prático ao contrário das áreas das humanidades ou de conhecimento puro como a física e a matemática.

2.2.2 Nível acadêmico Lato Sensu

Bem diferente do *Stricto Sensu*, a Pós-Graduação *Lato Sensu* tem como premissa inserir-se em uma perspectiva de educação continuada e de aprimoramento de conhecimentos. Com caráter de especialização em determinada área é uma ferramenta de aprimoramento de qualidades profissionais (Oliveira 1995 (18)).

Os cursos *Lato Sensu* apresentam-se como consequência da necessidade de ampliação de conhecimento que é impossibilitado durante a graduação, seja por sua carga horária, seja pelo seu planejamento curricular. Assim a especialização torna-se necessária como mais uma peça na engrenagem de aumento de conhecimento e especialização em uma área.

Entretanto no seu conceito esse nível de estudos não se limita somente a especialização de uma área, mas também provê uma forma do profissional se atualizar, promovendo acesso à conhecimentos que necessitam de revisão continuada.

2.2.3 Crescimento da Pós-Graduação Brasileira

São vários os fatores que propiciam o crescimento ou declínio do contingente de alunos em determinada área do conhecimento que são observados nos cursos de Pós-Graduação. Entre esses fatores alguns mais importantes são aqueles relacionados à ordem econômica, práticas que deixaram de ser utilizada ou mesmo pelo apelo das novas áreas.

Segundo Silva (2015) (19), observa-se que o diploma universitário em nível de graduação parece não mais garantir o favorecimento em sua entrada no mercado de trabalho. A vantagem que muitos veem antes de entrar em um curso de graduação, parece não ter o mesmo valor quando se sai das portas de uma universidade.

Para Pimentel (2017) (20), também a situação de “estar estudando” causa um impacto interessante nos jovens que pretendem continuar estudos após a graduação. Esse aspecto minimiza o impacto negativo de não ter encontrado um trabalho após a conclusão dos estudos em um mercado de trabalho cada vez mais competitivo e exigente nas competências de seus empregados.

No Brasil, entretanto, existem iniciativas governamentais importantes e que trabalham de forma ativa a partir de políticas públicas voltadas ao nível superior que vão ao encontro dos anseios desses jovens pesquisadores. Nos últimos anos também se observam o incentivo na

forma de bolsas de auxílio o que culminou em um alto índice de doutores e mestres titulados (Silva 2015 (19)).

Esse trabalho enérgico levou o Brasil a 13º. posição no ranking mundial e alcançando 2% da produção de artigos indexados. Um dado importante foi o salto realizado entre 1993 e 2013 que segundo Lievore (2017) (21) aumentou em 700% a produção de artigos científicos.

Já Guimarães (2013) (22) em seus estudos asseverou que a produção cientifica atingiu uma taxa de crescimento na ordem de 10,7% ao ano o que em termos práticos demonstra um aumento cinco vezes maior que a média mundial. Esse crescimento só foi possível porque as agências de promoção e financiadoras de projetos de pesquisa atuaram de forma sistematizada e obtiveram resultados excepcionais em suas ações.

As implementações significativas de projetos incentivados por políticas públicas é o que permitiu esse crescimento de forma consistente e com os resultados apresentados muito favoráveis (Lievore (2017) (21)).

2.3 A importância da produção cientifica

Se por um lado a Pós-Graduação no Brasil tem se mostrado um fator importante para o desenvolvimento tecnológico, econômico e social, o resultado prático dessa iniciativa tanto em instituições privadas quanto públicas resultam em uma séria de publicações científicas que denotam a responsabilidade de tais organizações e o retorno e dos recursos financeiros empregados.

Assim, indicadores que apontem a qualidade das publicações são fatore importantes para determinar se os resultados científicos estão sendo atingidos.

2.4 Arquiteturas para soluções tecnológicas de software

2.4.1 Processos de Desenvolvimento De Software

Este capítulo se refere às normas, processos e métodos utilizados durante o desenvolvimento de um aplicativo seja ele desktop ou mobile. Contudo está longe do escopo desse trabalho, uma dissertação mais aprofundada a respeito de todos os elementos que compõe todos os processos de engenharia de software, retingindo-se ao que foi aplicado no processo de desenvolvimento de software relacionado com essa pesquisa, limitando-se a apresentar os modelos clássicos da literatura.

Com o atual advento da internet, as aplicações se tornaram cada vez mais complexas e demandam das empresas cada vez mais esforços para tornar tais sistemas confiáveis e robustos.

A quantidade de informação que é disponibilizada demanda também sólida capacidade de armazenamento e segurança fazendo com que a mão de obra seja cada vez mais capacitada. Tudo isso faz com que o desenvolvimento de um novo software ou a manutenção do legado, deva seguir processos rígidos ou modelos de desenvolvimento sustentáveis que determinem sua continuidade em um mundo mutável como o atual.

2.4.2 Engenharia aplicada no desenvolvimento de software

Engenharia de Software (ES) é um conjunto de conhecimentos, organizados e elaborados com o intuito de descrever o instrumental técnico utilizado durante a construção de um programa de computador que compreende desde o a avaliação econômica, que é aquela que verifica se é economicamente pertinente a construção do código que o compõe, até a organização de controles e métricas que medem o esforço de elaboração e finalização (Resende (2005) (23)).

Para Resende (2005) (23) é uma metodologia que compreende a construção de sistemas baseados em partes construídas em módulos, estruturado em um processo dinâmico, rotineiro, inteligente e que atende a requisitos estabelecidos e com partes integradas funcionando de forma orquestrada e atendendo a padrões de qualidade fundamentados em tecnologia disponível. A orquestração se refere à integração dos fragmentos de software funcionando de forma organizada e é uma abstração de uma orquestra onde todos os instrumentos funcionam afinados e de maneira harmônica durante uma apresentação musical.

Ainda a engenharia de software deve envolver e ser conexa a variáveis intrínsecas a cada negócio assim como deve manter coerência em conhecimentos científicos e empíricos. Ela usa resultados das Ciências da Computação para obter os resultados pertinentes.

Segundo Pressman (2016) (24), a disciplina abrange os seguintes elementos fundamentais: métodos, ferramentas e procedimentos. Dessa forma os métodos determinam o "como fazer" para se alcançar e êxito no desenvolvimento. Já as ferramentas estão relacionadas com o apoio automatizado ou semi-automatizado e totalmente relacionado com os métodos. Por fim, como elo de ligação entre os métodos e ferramentas está os procedimentos que em seu cerne está alojado o resultado no formato de produção de um software com qualidade.

É notório que na atualidade o processo de desenvolvimento de software cresce a cada dia e novas técnicas são criadas em um processo acelerado uma vez que a sociedade *hight tech* toma cada vez mais espaço.

Observamos que desde a década de 1950, o que hoje está disponível com poder computacional atual nas mãos de muitos era impensável naquela época. Foi necessário um enérgico trabalho para compor metodologias de qualidade que assegurassem e desse suporte ao atual software e assim foram-se criando os atuais paradigmas de ES.

2.4.3 Paradigmas de Engenharia de Software

Devido à complexidade que os programas de computador estão hoje sendo desenvolvidos um dos desafios tanto dos engenheiros de software quando aos programadores, é trabalhar sobre uma perspectiva que propõe metodologias tanto para a parte da engenharia quanto para o gerenciamento dos projetos de software.

Apesar de existirem muitos paradigmas, e cada um aderir uma natureza de projeto específica, existem três principais etapas que são seguidas por todos os padrões (Pressman (2016) (24)):

- Levantamento de requisitos / Elicitação de requisitos
- Projeto e desenvolvimento
- Implantação / Manutenção.

2.4.4 Levantamento de Requisitos

O levantamento de requisitos é a forma em que um analista de software descreve, desde o entendimento da necessidade do usuário até o detalhamento das características funcionais e não funcionais do de um sistema e as regras de negócio que estão no escopo do projeto.

Os requisitos funcionais referem-se aos recursos que são perceptíveis pelo usuário após a implantação do projeto ou parte deste enquanto os requisitos não funcionais são aqueles que não estão na definição do escopo e do problema que o software irá resolver, mas trabalha em plano de fundo ou segundo plano, oferecendo a infraestrutura interna do software ou do ambiente operacional do mesmo.

Figura 3: Estrutura básica de requisitos de software. Fonte: próprio autor. São Paulo, SP. 2019

Os requisitos, representados na Figura 3, ainda estipulam os limites ou regiões em que o software deve operar, dentro de uma organização ou instituição traçando do que deve fazer parte do escopo do trabalho. Também determina restrições que tanto podem ser de negócios quanto técnicas (Pressman (2016) (24)).

2.4.4.1 Projeto / Desenvolvimento

Essa é a fase do processo de desenvolvimento de software onde é realizado as estratégias de técnicas, traçando um plano para atender aos requisitos definidos na etapa de Requisitos de Software, levando em consideração os Requisitos Funcionais, Requisitos Não Funcionais e Regras de Negócio. Nessa fase o analista de software irá decodificar o entendimento do processo feito até agora e elaborar um conjunto de técnicas aderentes às metas do projeto

2.4.4.2 Implantação e Manutenção

Durante a fase de implantação do sistema podem ocorrer diversas falhas oriundas da complexidade do ambiente de implantação. Dessa forma a implantação é uma parte importante

do processo onde o analista especifica onde e como o software será instalado e documenta o processo para que seja replicável em um momento futuro ou no caso de uma falha.

A manutenção está diretamente relacionada com a implantação pois ela é responsável por manter o software estável e disponível para os usuários. Observa-se então que os processos que compões as etapas básicas para desenvolvimento de software formam um ciclo que pode ser descrito na Figura 4. Setas indicam que os passos previstos não têm um fim definido. Após a finalização, partindo do levantamento e terminando na manutenção, o ciclo irá reiniciar indefinidamente para atender ao projeto atual como a novas características que poderão ser agregadas no futuro.

Figura 4: Ciclo básico do processo de software. Fonte: próprio autor. São Paulo, SP. 2019

A proposta Representada na Figura 4, apesar de aceita como um ciclo básico para o desenvolvimento ainda temos os modelos mais completos e tradicionais na área de ES.

2.4.5 Modelo Cascata (Waterfall)

O Modelo Cascada ou Waterfall é o modelo clássico em engenharia de software. Definido na década de 1970 por Winston Walker Royce (1970) (25), um cientista da computação (Royce 2018 (26)), esse modelo tornou-se a base para a criação de novos modelos de desenvolvimento. Tem como característica principal a relação dos processos de desenvolvimento organizado em fases e na forma sequencial "*Top-Down*".

Sua rígida estrutura de sequência determina que o próximo passo só pode ser seguido se cumprido todos os requisitos da etapa anterior. Isso torna esse modelo nos dias de hoje apenas uma referência. A complexidade, robustez e a constante mudança dos atuais sistemas computacionais demandam atividades paralelas que contempla as fases desde a elaboração dos programas até a fase final de implantação e manutenção.

O Modelo Cascata não determinou um padrão nas fases que compõe sua sequência. Assim é comum vermos o gráfico de etapas ligeiramente diferente, embora não altere o núcleo que compõe as etapas para alcançar os objetivos propostos do software (Pressman 2016 (24)).

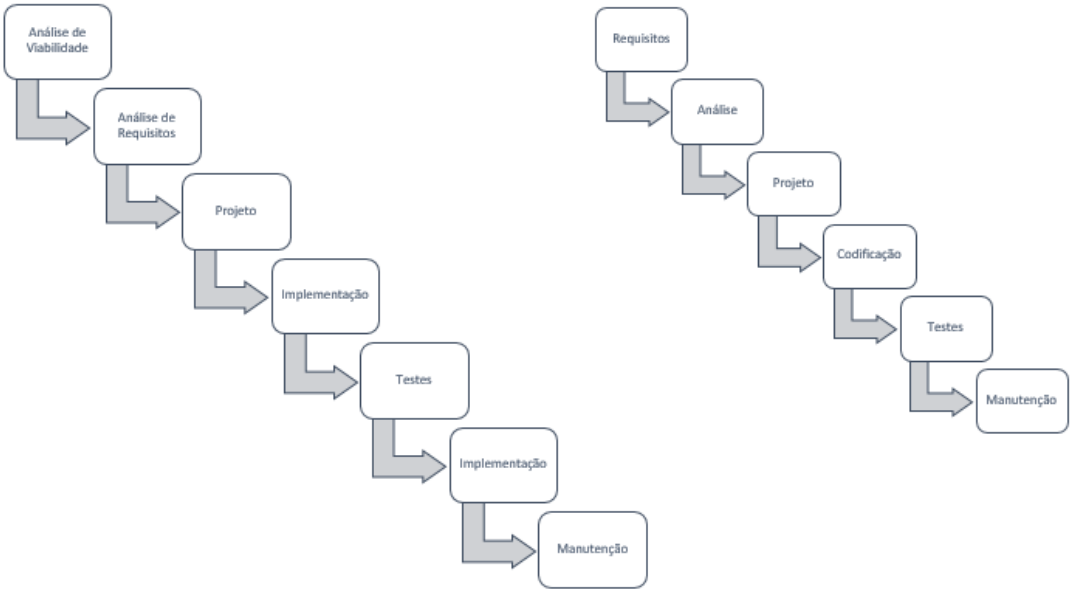


Figura 5: Modelo Cascata com diferentes fases e nomenclatura. Fonte: próprio autor. São Paulo, SP. 2019

A Figura 5 demonstra o ciclo de vida do modelo cascata com duas distintas nomenclaturas e fases, mas que em seu núcleo não perde a semântica do modelo e atingem o mesmo objetivo.

Apesar de se ter algumas vantagens importantes como a validação financeira do produto gerado em cada etapa, permitindo ao responsável e aos demandantes do projeto uma clara visão de custos do projeto, essa forma de conduzir o projeto enfrenta dificuldades na medida em que o processo não pode ser revisto nas etapas anteriores.

Foi proposto então uma modificação no modelo clássico chamado de Modelo Cascata Revisto (Pressman 2016 (24)) onde é possível agora retroceder etapas e corrigir falhas nas tarefas de etapas anteriores. Sem essa mudança, havia um risco muito grande de haver um ciclo interminável elevando os custos do projeto podendo até inviabiliza-lo.

Para cada etapa do Modelo Cascata foram definidas tarefas para se alcançar os objetivos. Como visto anteriormente o modelo não tem um padrão rígido da sequência de etapas. Abaixo estão elencadas as definições de algumas mais utilizadas em projetos que utilizam essa forma de desenvolvimento (Pressman 2016 (24)).

2.4.5.1 Fase de Requisito

Essa normalmente é a fase inicial da maioria dos projetos que se utilizam desse modelo para o desenvolvimento de software. Outros projetistas de software ou designs de projetos iniciam o processo com a análise de viabilidade que está intimamente ligada com os custos financeiros. Mas via de regra os requisitos serão sempre os primeiros contextos de engenharia para se iniciar os projetos (Resende 2005 (23)).

Nessa fase é criada a comunicação entre os responsáveis pelo projeto e os investidores e delimita as exigências, as obrigações e premissas que circundam o projeto. Aqui são definidos os tipos de serviços que o software deve atender, suas limitações e seus objetivos para o negócio da empresa ou instituição. Normalmente está relacionada aos requisitos funcionais do sistema, ou seja, aquilo que será perceptível pelo usuário ao utilizar a aplicação e os resultados providos pela interação entre o usuário e a máquina.

De forma geral essa fase é escrita e estabelecida na forma de um contrato por analistas de negócios que serão responsáveis depois para transferir e traduzir o conhecimento para que os engenheiros de software possam atuar.

2.4.5.2 Fase de Projetos

Essa fase é elaborada com processos que caracterizam a forma em que o sistema será construído. Assim observamos que há definições que vão desde a estrutura em que os dados são organizados, a organização das interfaces de comunicação.

Em linhas gerais é definido a arquitetura de software que será implementada e a plataforma em que será disponibilizada

2.4.5.3 Fase de implementação

Esta é a fase em que o sistema é organizado em termos de código fonte. A implementação nada mais é que a codificação do sistema. Tem como base o documento de requisitos que explicita a forma que o sistema de computador deve se comportar e quais são os produtos esperados durante a operação deste (Resende 2005 (24)).

Espera-se que nessa fase o nível de detalhamento das peças de software sejam elevadas, isto é, a codificação deve levar em consideração diversos aspectos que são características da tecnologia que será utilizada para que seja alcançado de maneira efetiva os objetivos do software.

Aqui o Engenheiro de Software irá aplicar conhecimentos técnicos, como a utilização de *Desing Patterns*[1], escolha da melhor arquitetura tanto de software quanto de hardware para melhor atender o domínio da aplicação, ferramentas de geração automática de software[2] que se baseia diretamente na documentação levantada na fase de requisitos (Valente 2008 (27)).

É importante ressaltar que testes unitários são realizados nessa fase. Testes unitários são pequenos testes, como por exemplo, a validação da entrada e saída de dados. A partir dessa observação, objetiva-se a mitigação dos erros que serão investigados na fase de testes.

2.4.5.4 Fase de integração e teste

Conforme Valente (2008) (27), a fase de integração é a agregação de todos os módulos, configuração de programas individuais em um ambiente comum para que sejam testados e avaliados como apenas sistemas que funcionem de forma independente. A importância dos testes é para garantir:

a) Que os requisitos de software foram atendidos
b) Verificar inconsistências nos dados gerados pela aplicação
c) Correção de erros de software (*bugs*)

2.4.5.5 Fase de operação e manutenção

A operação refere-se a quanto o sistema é instalado e colocado em estado operacional em um ambiente computacional compatível com os requisitos de software e acessível ao usuário. É nesse momento que a aplicação é entregue ao usuário.

A manutenção tem dois aspectos principais. Corrigir falhas de erros que não foram detectados na fase de testes e a inclusão de novas características no software. Muitas vezes a inclusão de novas características irão necessitar de um novo projeto, pois a modificação estrutural e de negócio pode mudar. Entretanto pequenos ajustes acontecem na fase de manutenção

[1] Desing Pattern são modelos ou forma de se desenvolver trechos de códigos de forma padronizada e baseada em boas práticas de programação (Pressman (2016) (24)).

[2] A geração automática de software se baseia em ferramentas que Engenheiros de Software utilizam para documentar o sistema. A partir do desenho das funcionalidades desses programas de computador, é então possível gerar código de forma automática que servirá como base para que os programadores possam iniciar seu desenvolvimento. Como o código gerado é baseado na documentação, fica reduzido os erros provenientes do não entendimento do domínio do problema . (Valente 2008 (27)).

2.4.6 Modelo Incremental

O modelo incremental é também conhecido como parte do grupo de modelos evolucionários. Essa denominação de evolucionário vem do fato de que a crescente demanda do mercado e a complexidade atual dos sistemas tem requerido uma forma mais expressiva e rápida de desenvolvimento e manutenção do software (Pressman 2016 (24)).

Ao contrário do modelo cascata que tem uma forma rígida em seu processo de passos, o modelo incremental combina elementos entre os fluxos dos processos podendo ter diversos processos paralelos e lineares ao mesmo tempo (Santos 2016 (28)).

No modelo incremental é utilizado o modelo cascata, mas de maneira iterativa. Isto é, ao invés de se apresentar o sistema como um todo, ele aplica as regras do modelo de cascata em pequenos blocos de características que são do domínio da aplicação. A ideia central é entregar produtos de características completas a cada novo clico.

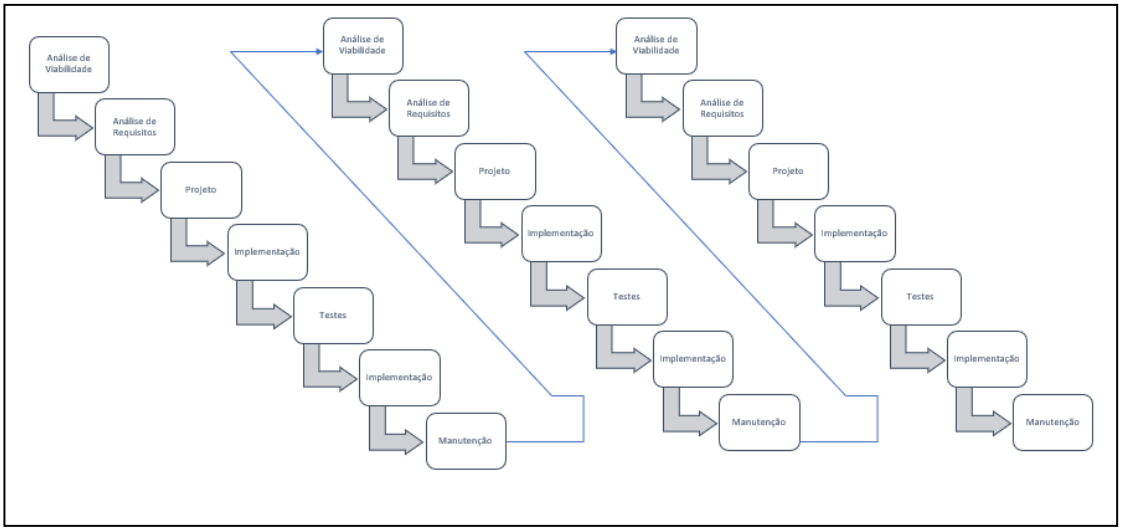


Figura 6: Iterações no Modelo Incremental. Fonte: próprio autor. São Paulo, SP. 2019

Esse modelo, Figura 6, é muito apropriado quando se tem as regras de cada elemento de características de software bem definidas e uma visão geral do sistema como um todo. Segundo Lessa (2015) (29), é útil também quando existe a necessidade de construção de um software complexo, mas não existe mão de obra em abundância. Aplica-se a regra “dividir para conquistar”, realizando o desenvolvimento de características completas e assim atendendo aos poucos uma demanda de software.

2.4.7 Paradigmas de programação

Para esse trabalho iremos apresentar dois paradigmas que são relevantes para alcançar os objetivos propostos os quais são o Paradigma Procedural por ser um clássico tanto na literatura e normalmente a porta de entrada de novos programadores e o Paradigma Orientado a Objetos no qual foi desenvolvido o projeto do presente escopo. Contudo existem outros paradigmas tão importantes em arquiteturas de sistemas os quais podemos citar: paradigma funcional, imperativo, etc.

Linguagens de programação sofrem muitas adições em suas funcionalidades ao longo do tempo para determinar sua estabilidade nos meios para que foi concebida (corporativa, educacional). Com a evolução da capacidade de processamento e o aumento de volume de informações que necessitavam de tratamento foram desenvolvidas ao longo das últimas décadas, estratégias de desenvolvimento e, portanto, do comportamento da linguagem em si.

Assim como os paradigmas de Engenharia de Software, existe uma classificação para determinar a forma das linguagens de computador em que é baseado suas funcionalidades.

Um paradigma de programação é então um arcabouço de conceitos organizados, de forma a dar ao desenvolvedor vantagens durante o processo de desenvolvimento, para que as organizações e instituições alcancem seus objetivos sejam estes corporativos ou institucionais.

Segundo Roy (2009) (30), é muito mais interessante para o desenvolvedor de software concentrar-se em entender como funcionam os paradigmas de programação do que com as linguagens pois há muito menos paradigmas do que linguagens.

Dessa forma, antes de se escolher a uma linguagem, deve-se primeiro entender a natureza do problema que se quer alcançar. O paradigma irá orientar sobre a melhor estratégia que atenderá ao objetivo proposto e só depois escolher a linguagem. Ainda assevera Roy (2009) (30), que por exemplo, linguagens orientadas a objetos existem muitas como Java®, Java®script, C++, C#, etc. Porém quem determina a utilização da orientação a objeto ou um desenvolvimento procedural é o problema e não a linguagem

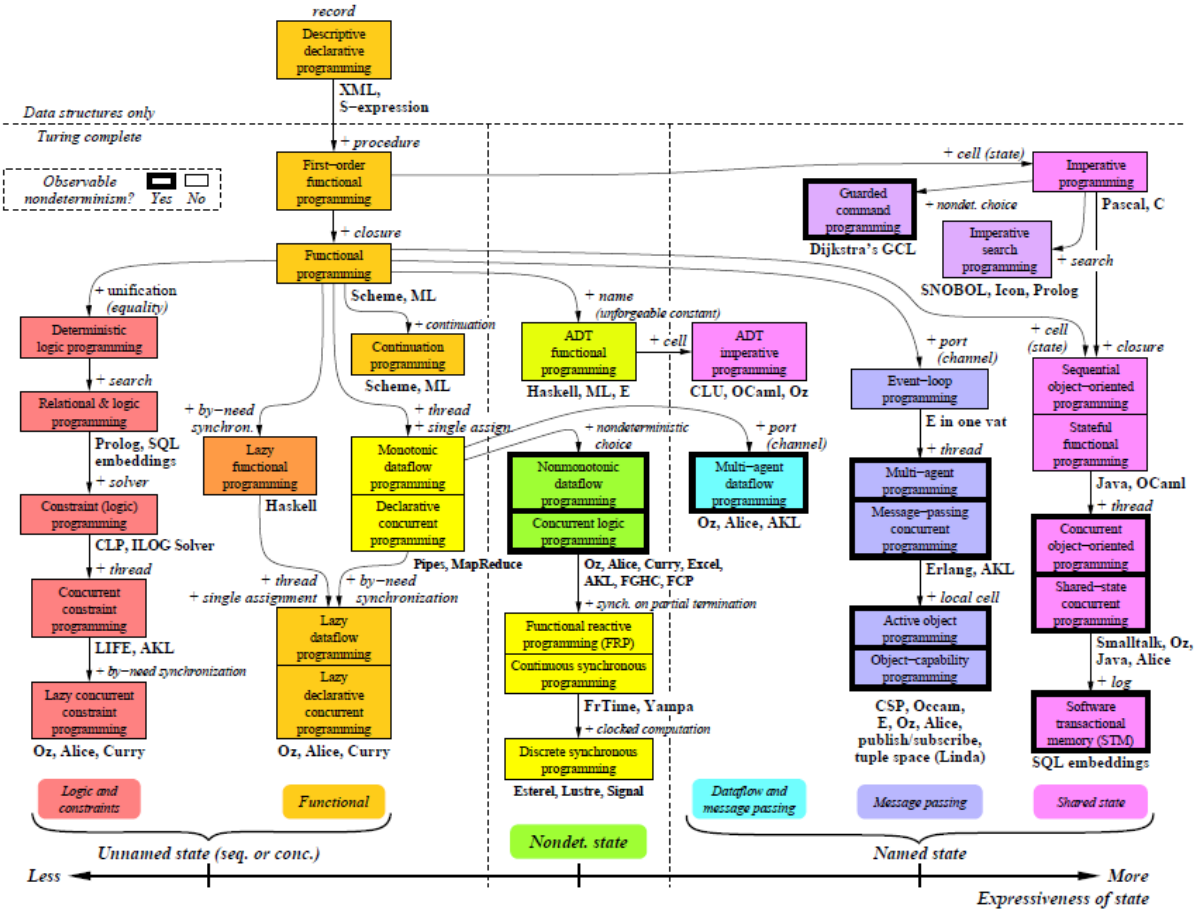


Figura 7: Taxionomia dos paradigmas de Programação : Fonte: Roy (2009) (30)

A taxinomia apresentada por Van Roy (2009) (30), Figura 7, é extremamente pertinente para a escolha de uma linguagem a partir de um paradigma. A Figura 7 mostra a organização das linguagens dependentes do paradigma escolhido.

2.4.8 Paradigma Procedural

O paradigma procedural pode ser classificado como um paradigma clássico. É muito comum ser apresentado para programadores iniciantes pela facilidade de aprendizado e útil para apresentar conceitos mais aprofundados da linguagem uma vez que possui poucas regras para sua programação. Esse paradigma também é conhecido como Programação Estruturada.

Conforme Biondo (2017) (31), o paradigma procedural é dividido em três regras principais:

- Sequência lógica
- Estruturas condicionais
- Laços de repetição

A sequência lógica é a ordem em que as regras são aplicadas durante o processamento do programa. A ordem lógica é importante pois o programa é executado sequencialmente e dependendo das regras de negócio uma regra X não pode ser executada antes da regra Y sem que haja erro de processamento. Em um exemplo fácil de entender, podemos imaginar o processo de transação bancária. Só é possível executar um saque se antes for verificado se existe saldo suficiente. Esse é um exemplo em que a ordem das regras deve ser obedecida sequencialmente.

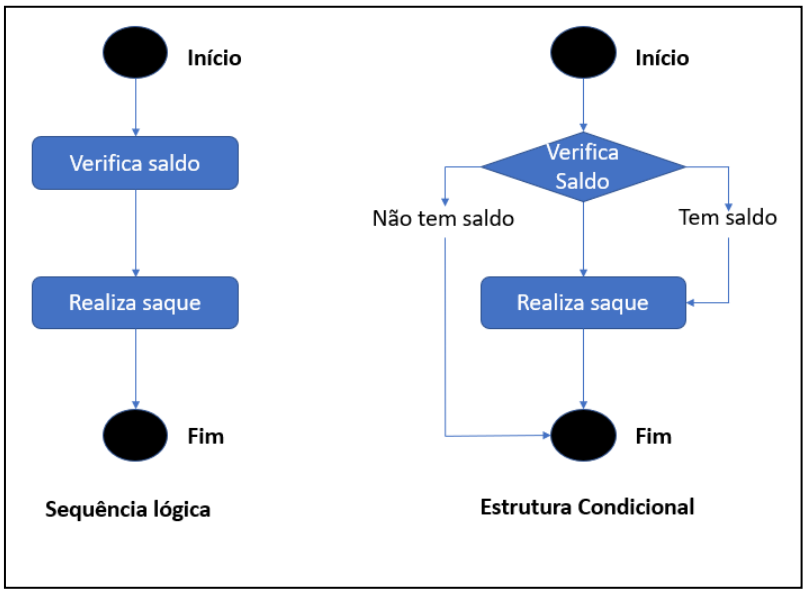


Figura 8: Fluxo de sequência lógica e laço condicional. Fonte: próprio autor. São Paulo, SP. 2019

Entretanto o encadeamento lógico não é suficiente para atender as necessidades de um programa robusto. As estruturas condicionais entram como a forma de resolver essa questão utilizando palavras da sintaxe da linguagem como *if* (se) else *if* (senão se), *else* (senão) e *switch* (Biondo 2017 (31)). Dessa forma os programas ficam mais seguros e é possível incluir regras que permitam ao computador tomar decisões assertivas antes de processar operações programadas (Deitel 2016 (32)).

Os laços de repetição são o que permitem criar rotinas automatizadas para grande volume de dados. Tais estruturas programáticas são ótimas para realizarem um conjunto de tarefas tantas vezes quanto forem necessárias, para um determinado processamento, e são definidas nas atuais linguagens de programação de alto nível como a sintaxe *Do While* (faça enquanto) e *While* (faça) (Deitel 2016 (32)).

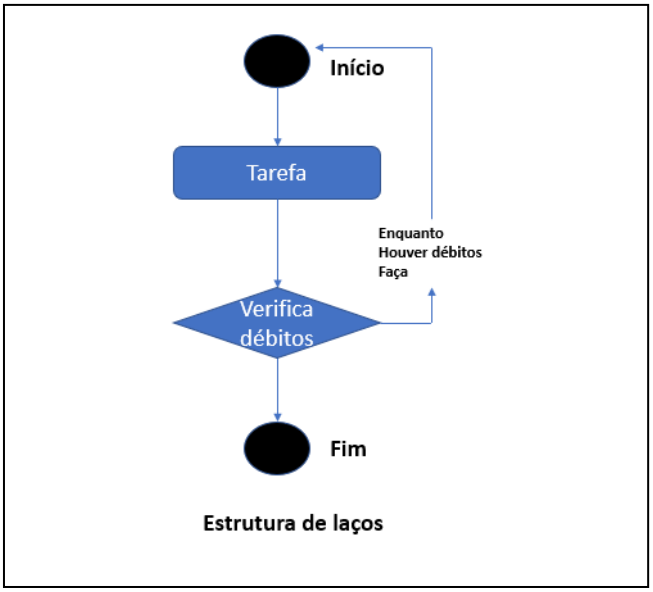


Figura 9: Fluxo de estrutura de laço. Fonte: próprio autor. São Paulo, SP. 2019

A sintaxe *While* diz ao computador realizar algo repetidas vezes até que uma condição seja estabelecida. Imaginemos o seguinte cenário: uma administradora de cartões de crédito irá lançar os débitos nos cartões dos clientes durante a noite em uma rotina. Pega-se a lista de débitos e para cada débito aplica-se este para o cartão correspondente. Essa é uma rotina típica de repetição. A rotina finaliza, quando todos os débitos forem lançados para os respectivos cartões.

A natureza em que se apresentava inicialmente o paradigma procedural e com a complexidade em que se tornaram os sistemas de informação atuais, levou a necessidade de novas técnicas para abordar os problemas emergentes (Biondo (2017) (31)). Essas técnicas têm como premissa a simplificação dessa abordagem de programação e posterior manutenção de tais programas.

Um conceito muito difundido no meio acadêmico que auxiliou na forma em que os programas de computadores diminuíssem sua complexidade é o conceito de “dividir para conquistar”. Esse conceito leva em consideração que dividindo o problema em partes menores, encontra-se então a solução de forma mais rápida e segura segundo Deitel (2016) (32);

Dessa forma pensou-se então na modularização dos sistemas e cada módulo é então responsável por apenas uma parte da aplicação. Além da modularização, o paradigma procedural organiza um programa em blocos de código e de variáveis globais tornando essa estratégia de programação útil para determinas finalidades de processamento.

2.4.8.1 Paradigma orientado a objetos

O paradigma procedural tem como base a execução de um conjunto de procedimentos e regras que seguem um fluxo contínuo na programação e que estão inter-relacionados. A característica Orientada a Objetos (OO) tem como premissa a correlação do mundo concreto, convertido em uma abstração programática e troca de mensagens entre os componentes.

Essa intersecção de elementos do mundo real trouxe características importantes na linguagem como os atributos, ou conjunto de características próprias de uma entidade e os métodos que são ações que acontecem no mundo real segundo Puga (2004) (33).

Um objeto é um conceito abstrato que levamos para o computador classificando-o conforme a necessidade da programação. Isto é, está nas mãos do desenvolvedor encontrar quais características do mundo real ele deve transportar para o programa de computador a fim de atender às necessidades do sistema. Ensina Puga (2004) (33), que o processo de abstração de um objeto do mundo real para a computação é observar os aspectos essenciais que definem um objeto e ignorar atributos que não são relevantes para uma modelagem computacional.

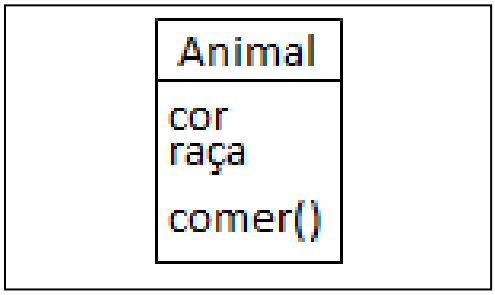


Figura 10: Exemplo de uma classe com atributos e métodos. Fonte: próprio autor. São Paulo, SP. 2019

Entretanto existem outros aspectos importantes que definem o paradigma orientado a objetos além da abstração. São eles o encapsulamento, o polimorfismo e a herança.

O encapsulamento é a capacidade de um objeto ser autocontido com propriedades ou atributos e métodos ou ações em uma única entidade. Assim o conteúdo de uma variável só é acessado através de métodos que forneçam acesso a ela permitindo que haja segurança ao conteúdo. Um exemplo típico é tentar adicionar uma data inválida no sistema. O método responsável pela adição dessa data poderá interceptar o conteúdo, validar o dado que será inserido e tomar a decisão se atualiza ou não. Imaginemos uma data onde o dia do mês é maior que 31. Dessa forma a regra de negócio não deve permitir uma inclusão como essa (Pressman (2016) (24)).

A herança cria uma hierarquia entre objetos e desse relacionamento emerge a capacidade de se reutilizar código já construído e testado para uma nova classe que tenha uma relação direta entre a classe Pai e a classe Filho. É a herança que permite que sejam compartilhados atributos e ações que poderão ser reutilizadas pela classe filha. O que torna a herança robusta é a capacidade não apenas de herdar ações e métodos, mas também poder especializar programaticamente essas ações para atender a necessidades especificas de uma classe filha ou subclasse (Deitel (2016) (32)).

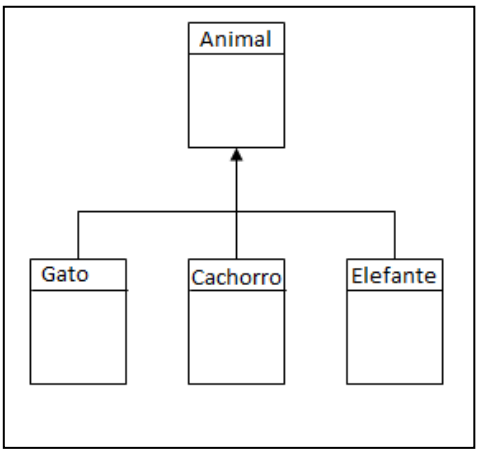


Figura 11: Típica hierarquia de classes. Fonte: próprio autor. São Paulo, SP. 2019

O polimorfismo é a capacidade de um objeto se especializar, conforme foi dito anteriormente. Mas vai além. Quando uma instancia de uma classe é filha de uma superclasse a variável utilizada na programação poderá recebe-la como valor utilizando técnicas de casting. Isso dá ao desenvolvedor uma forma de criar programas que recebe dados de diversas fontes e trata-los de forma diferente na medida em que uma instancia de classe se apresenta no fluxo normal do programa pode ter diversas formas.

A conjunção de objetos trabalhando juntos em total sincronia é o que dá poder às linguagens de programação que utilizam essa técnica de desenvolvimento. A Programação Orientada a Objetos (POO) tem como principal abordagem em seu fluxo a comunicação entre objetos e troca de mensagens.

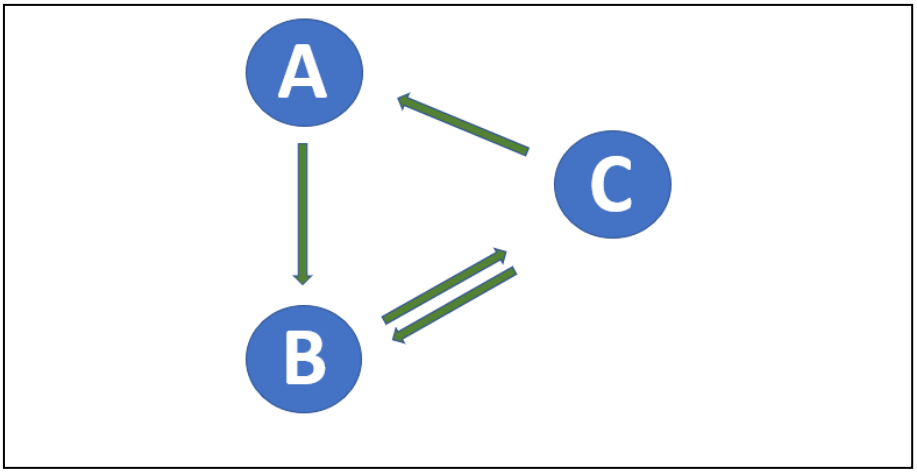


Figura 12: Comunicação entre três objetos. Fonte: próprio autor. São Paulo, SP. 2019

A Figura 12 ilustra um típico modo de comunicação entre os objetos em tempo de execução. Podemos observar que o objeto A manda mensagens para o objeto B. Mas B não pode se comunicar com A enviando mensagens. Entretanto B envia e recebe mensagens de/para C. Por sua vez C manda mensagens para A, mas não recebe de A alguma mensagem.

Essa orquestração de troca de mensagens é o núcleo do POO. Cada elemento tem seus atributos e ações ao mesmo tempo cada entidade funciona como uma peça de uma grande engrenagem. A troca de informações entre esses elementos torna essa técnica de desenvolvimento ótima os sistemas atuais pois permite maior segurança e escalabilidade da aplicação.

2.5 XML – eXtensible Markup Language

Esse capítulo apresenta o que é o XML e como ele é representado. A importância das linguagens de marcação vem do fato delas serem consideradas uma forma organizar e mapear a informação na internet para que ela possa ser lida por sistemas de computador e a partir disso compilar, organizar, processar e sumarizar os dados para posteriormente serem transformados em informação útil para as organizações.

A informação como está distribuída hoje ainda não permite uma forma fácil de organizar os dados. Mas grandes empresas, como o Google®, Bing®, Microsoft®, etc, estão fazendo esforços para chegar a um nível de padronização dos dados que estão distribuídos em diversas plataformas e formatos sejam lidos pelas máquinas e trazer valor para os dados armazenados. Nesse contexto o XML é uma ótima alternativa, pois sua estrutura permite organizar os dados facilitando a leitura, decodificação e geração de conhecimento para tomada de decisão.

2.5.1 XML

O XML (eXtensible Markup Language) é uma linguagem descritiva. Isso significa que ela permite dar interpretação a informação auto-contida. Dessa forma quando se diz "linguagem", esta não está se referindo a uma estrutura lógica que é o núcleo das linguagens de programação para computadores, mas sim a *tags* ou marcações que proveem significado e contexto ao fragmento de texto a que se refere.

Assim, segundo o W3C (2018) (34) (World Wide Web Consortium) o XML é uma linguagem que descreve como programas de computador podem realizar a leitura dos dados assim como uma linguagem descritiva de classe de objetos de dados.

Ela é também considerada uma especialização ou um subconjunto da SGML (Standard Generalized Markup Language), ou seja, é uma especificação ou padronização geral para linguagens de marcação.

Uma marcação XML é composta de um rótulo que identifica a semântica do dados e de alguns símbolos que conjuntamente permitem ao computador entender e classificar a informação fazendo com que as partes do texto sejam perceptíveis e distintas do resto do documento dando ao computador a capacidade de analisar adequadamente cada bloco de texto Ray 2003 (35).

```
<?xml version="1.0" encoding="UTF-8" ?>
<curriculo_lattes data_processamento="04/05/2017 14:41:30">
  <pesquisador id="3089430786971948">
    <identificacao>

      <identificadorl0></identificadorl0>
      <nome_inicial></nome_inicial>
      <nome_completo></nome_completo>
      <nome_citacao_bibliografica></nome_citacao_bibliografica>
      <sexo></sexo>

    </identificacao>
    <idiomas>

      <idioma>
          <nome></nome>
          <proficiencia></proficiencia>
      </idioma>

      <idioma>
          <nome></nome>
          <proficiencia></proficiencia>
      </idioma>

    </idiomas>
    <endereco>

      <endereco_profissional></endereco_profissional>
      <endereco_profissional_lat></endereco_profissional_lat>
      <endereco_profissional_long></endereco_profissional_long>

    </endereco>
```

Figura 13: Fragmento de XML gerado pela ferramenta LattesXtractor. Fonte: próprio autor. São Paulo, SP. 2019

A Figura 13 apresenta um fragmento de documento XML e demonstra seu potencial para extração dos dados de forma estruturada onde tanto um humano quanto a máquina pode entender o significado de seu conteúdo.

Ela primeiro define a versão do XML e o tipo de codificação dos caracteres usados no texto para que o interpretador ou *parser* de XML possa extrair os dados de forma eficiente e correta da informação contida.

Todo o conteúdo textual fica envolvido entre duas marcações ou tags, estruturando dessa forma o conteúdo. Uma marcação é uma delimitação de onde se inicia e onde termina o texto em questão com alguns símbolos complementares.

Por exemplo, toda marcação inicia com um símbolo de menor “<” e termina com um símbolo de maior “>” e é envolvido por um texto de denota o tipo do conteúdo. Um exemplo dessa marcação é a “idioma” com a marcação “<idioma>”. Para determinar o fim da marcação o símbolo de barra “/” é acrescentado a marcação ficando com a tag “</idioma>”. Todos o conteúdo textual entre essas duas marcações é que pode ser extraído de informação útil.

Essa forma de marcação permite que sejam criados modelos de XML para qualquer tipo de aplicação (Fugeri 2001 (36)) tornando o documento capaz de reconhecer e apresentar o significado da informação.

Ainda segundo Fugeri (2001) (36), essa codificação do conteúdo textual permite que ferramentas computacionais podem agora se adaptar facilmente à nova realidade de extração de informações ou conhecimento de forma segura, permitindo assim análises complexas de conteúdos textuais. Existem ainda outras informações úteis no XML como as propriedades. Estas são colocadas no corpo da tag permitindo incluir mais informação. Por exemplo, a tag "<idioma>" poderia ter uma propriedade como código da língua e sua notação seria algo como "<idioma cod_lingua='pt'>Português</idioma>"

2.5.1.1 DOM – Document Object Model

A natureza de estrutura de um arquivo XML remete diretamente a uma estrutura de árvore em forma hierárquica. Esse formato organização dos dados permite que sejam manipuladas as informações através da leitura de forma simplificada utilizando DOM – Document Object Model.

Segundo Siqueira (2003) (37), através de uma estrutura DOM é possível "*processar, criar e editar documentos XML, utilizando linguagens de mercado*" e atuais. Com um arquivo XML bem formado, isto é, sem erros em seu corpo de conteúdo é possível extrair dos dados através de funções simples já implementadas nas diversas linguagens de programação.

Então extrair o conteúdo de uma marcação como a "<idiomas>", uma forma é chamar uma função como "getElementByTagName('idiomas')"[3] e o conteúdo é extraído automaticamente podendo retornar uma simples cadeia de caracteres para o programas ou uma lista de cadeia de caracteres, caso exista mais de um idioma definido dentro da marcação "<idiomas>".

[3] getElementByTagName é apenas um exemplo na linguagem javascript de como se extrair informações de uma árvore DOM. Cada linguagem de programação implementa a recuperação da informação da forma que acha mais adequada.

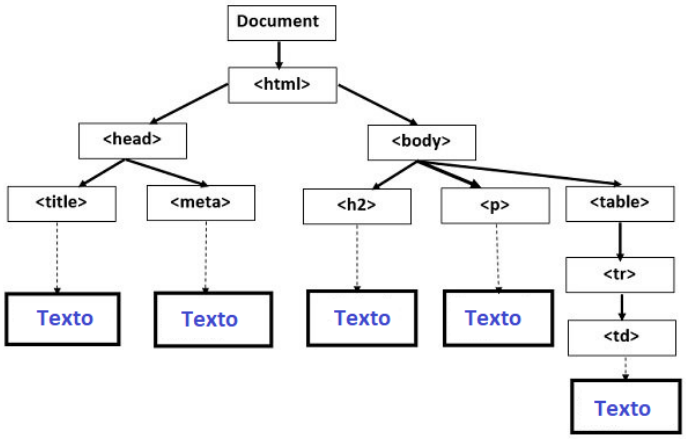


Figura 14: Um exemplo de árvore DOM a partir de um documento HTML. Fonte: próprio autor. São Paulo, SP. 2019

A Figura 14 demonstra uma árvore DOM criada em memória do computador como resultado do processamento de um documento XML criado a partir de um arquivo HTML (*Hipertex Markup Language*). O programa irá ler o arquivo XML em formato textual, transformar sua saída, através de um *parsing* de XML, em um conteúdo em memória e recuperar o conteúdo de texto com apenas uma chamada de método da linguagem utilizada.

Importante notar que todo o conteúdo extraído é um uma "folha" na estrutura de árvore, isto é, depois desse elemento não há mais elementos. Outro ponto importante é que o algoritmo de recuperação de conteúdo de um HTML exige uma ordem. Primeiro são extraídos os conteúdo que estão à esquerda e depois os da direita, chamada de pré-ordem ou quando ao contrário, ou seja, primeiro os elementos da direita e depois o da esquerda, é chamado de pós-ordem.

Então para extrair o texto que está na marcação "<head>" primeiro ele procurará na marcação "<title>" e depois na marcação "<meta>". Essa é uma regra para leitores de conteúdo HTML. Mas cada aplicação utiliza o conteúdo textual da forma que achar mais conveniente. No caso do HTML essa ordem é para que qualquer navegador da internet *renderize* (apresente) o conteúdo na tela.

2.5.1.2 DTD – Document Type Definition

Apesar da facilidade de criação de um arquivo XML com marcações personalizadas que atendam às regras de uma determinada aplicação existem casos em que há necessidade de se obter regras que validem o tipo de dado que está sendo enviado para aplicação.

Um exemplo típico poderia ser a consulta de dados de um objeto postado e disponível para consulta no site dos Correios. Supondo que uma determinada consulta leve em consideração o campo CEP, pode-se criar uma DTD (*Document Type Definition*) que identifique se o dado enviado é composto somente por números com sete dígitos que é a regra do código postal brasileiro atualmente. A importância vem do fato de que ao validar a informação mitigaria por exemplo a invasão do site dos Correios com o envio de um dado inconsistente trazendo segurança às empresas e uma programação mais segura.

Para situações hipotéticas como essa descrita acima é que foi criado a DTD, isto é, um mecanismo para verificar se o XML está no formato esperado e correto através do analisador sintático do *parser* de XML. A DTD também determina quais são as marcações e atributos permitidas (Santos 2017 (38)).

Sobre a validade de um documento XML, existem duas situações que são avaliadas pelo *parser* que utiliza um DTD que retornam ou não a invalidade do documento:

a) Um documento XML que não está implementado com todas as regras do DTD, mas está em sua forma sintática correta, então pode-se dizer que é um documento XML bem formado, mas não é válido. Dessa forma o *parser* irá responder com uma mensagem de erro típica para documentos bem formados, mas inválidos;
b) Um documento XML que não está implementado segundo as regras do DTD é um documento inválido.

2.5.1.3 Estrutura básica de um DTD

Para declarar uma estrutura DTD, ele dever ser definido no início de um arquivo XML logo após a declaração “xml version”ou através de um arquivo externo. No caso de um arquivo externo o arquivo pode ser local (na máquina em que o programa está rodando) ou em um site na Internet (Groner (2009) (39)).

Para a declaração que está incluída no arquivo XML temos a sintaxe:

<!DOCTYPE elemento-raiz [

Declaração-de-elementos

]

Abaixo segue um exemplo de um arquivo XML completo com sua declaração DTD:

```
<?xml version=”1.0”>
<!DOCTYPE libray [
    <!ELEMENT libray (autor_name, book_name, year_of_publication)
    <!ELEMENT autor (#PCDATA)>
    <!ELEMENT book_name (#PCDATA)>
    <!ELEMENT year_of_publication (#PCDATA)>
]>
<library>
    <autor_name>        </autor_name>
    <book_name></book_name>
    <year_of_publication></year_of_publication>
</library>
```

A estrutura do arquivo DTD, conforme foi dito anteriormente pode ser local (na máquina do usuário), no próprio arquivo XML (como do exemplo acima) ou em um arquivo na internet.

Para um DTD na máquina local utiliza-se a directiva SYSTEM e o caminho do arquivo DTD. Por exemplo:

```
<!DOCTYPE library SYSTEM “my_library.dtd”>
```

diz ao parser que o arquivo está na mesma pasta que o arquivo XML com o nome “my_library.dtd”. Esse é um arquivo de texto comum que dentro contém as definições.

Para denotar um arquivo disponível na internet, a sintaxe de definição da DTD é apresentada:

```
<!DOCTYPE library SYSTEM “my_library.dtd”[
    “http://www.<dominio_internet>.com/my_library.dtd”
]>
```

Todas as declarações elementos vem na forma da declaração ELEMENT:

```
<!ELEMENT myElement (#PCDATA)>
```

Há uma série de regras que definem o formato de um arquivo XML e dessa forma determinam como o arquivo deve ser lido pelo analisador sintático.

2.5.2 Processamento de XML utilizando bibliotecas Java®

O XML é um documento em texto que precisa ser processado por um motor eficiente e que traga a facilidade para manipulação da informação armazenada nas tags (marcações). A linguagem Java® possui diversos motores com *drivers* (controladores) com características que atendem aos diversos tipo de aplicação. Dentre as mais conhecidas APIs estão a biblioteca SAX (*Simple API for XML*) e a biblioteca DOM4J. (*Documento Object Model for Java*) (Dev Media (2012) (40)).

A biblioteca SAX realiza a leitura de um arquivo XML e dispara eventos na medida em que encontra tags. Essa interface facilita para o programador a recuperação de um dado em um arquivo XML de maneira fácil. Cada evento disparado tem um método associado o qual o desenvolvedor implementa e registra no motor do parser. Essa técnica permite uma arquitetura onde o custo com o armazenamento em memória seja baixo para o hardware, porém utiliza mais tempo de entrada e saída (I/O).

A biblioteca DOM4J, permite o acesso aos dados através da leitura armazenada em forma de árvore binária na memória do computador. Essa característica implica na recuperação do dado em alta velocidade, pois o XML é lido por completo e a recuperação se dá através do acesso ao dado na memória. A desvantagem dessa estratégia é que para XML longos, esta técnica fica inviável pelo alto custo de armazenamento interno em memória.

2.6 Linguagem Java®

Java® é uma linguagem de computador, *open source*, orientada a objetos, criada inicialmente pela empresa Sun Microsystems© sendo a Oracle© atualmente a detentora de seus direitos autorais. Baseada em C++ em sua estrutura de programação, eliminou as principais características entendidas como difíceis de se programar (Claro (2000) (41)).

Um exemplo da complexidade removida da linguagem é o uso de ponteiros, muito utilizada na linguagem C/C++, recurso esse muito útil ao desenvolvimento de sistemas complexos, entretanto dispensável nos ambientes operacionais em que sistemas exigem mais do negócio e mais simplicidade na codificação, entretanto perde para sistemas de alta performance e também de sistemas que necessitem da herança múltipla, outra característica poderosa, mas que causa muita confusão para os desenvolvedores.

Atualmente Java® é utilizada para desenvolvimento desktop, mobile e dispositivos de baixa capacidade de processamento. Sua versatilidade compreende desde aplicações comerciais como aplicações mais específicas como computadores de bordo de automóveis, celulares, cartões com chip que exijam processamento. Entretanto é percebido sua presença principalmente em sistemas web.

2.6.1 Histórico da linguagem Java

Na década de 90 o cientista da computação, James Grosling especificou a linguagem de programação, chamada Oak, onde o time de desenvolvimento acreditava que seria o próximo salto da era dos computadores. O Green Team (Cavalcanti (2016) (42)), como era chamado o grupo que compunha o *Project Green*, trabalhava em tecnologias de empresas ligadas ao ramo de eletrônica e tinham como objetivo levar a inteligência de software para equipamentos de baixo consumo (Claro (2000) (41)) de memória, como por exemplo, o controle remoto de TVs a cabo, mas percebeu prematuramente que esse público não teria um campo de atuação viável economicamente.

Durante esse tempo ainda de instabilidade onde não se sabia exatamente onde encaixar a linguagem, observaram que era impraticável criar uma versão para cada tipo de dispositivo. Foi desenvolvido então um sistema operacional chamado GreenOS, que utilizava a linguagem Oak para resolver esse problema. Agora a linguagem Oak tinha um ambiente para rodar nativamente.

Conforme Cavalcanti (2016) (42), a equipe do Project Green não conseguia um contrato para que o desenvolvimento da linguagem patrocinado e ainda sem grandes atenções por parte das empresas que desenvolviam hardware na época. Entretanto em 1995 com a explosão da internet a equipe de desenvolvimento conseguiu realizar um contrato com a antiga Netscape dona do navegador de mesmo nome e levar mais interatividade baseada em programação para o que foi chamado de início da era internet.

A linguagem foi renomeada para Java® por uma necessidade patente. Já havia uma linguagem de nome Oak registrada. Fato curioso é que a equipe de desenvolvimento consumia muito café em uma cafeteria próxima a Sun Microsystens, quando estava reunida. Os cafés da ilha de Java, uma pequena ilha da Indonésia eram os mais caros do mundo e também eram os preferidos pelos desenvolvedores. Em reunião com a equipe de marketing, acharam então

apropriado o nome Java como nome da linguagem e esta se tornou uma das mais importantes linguagens atuais.

O kit de desenvolvimento de aplicações Java® é fornecido de forma gratuita pela atual Oracle e é denominado Kit de Desenvolvimento Java® (JDK). Em seu núcleo estão as principais ferramentas para o desenvolvimento e é dividido em 3 edições principais:

2.6.2 Java® Standard Edition (JSE)

Representa a núcleo do Java®. Essa edição ou módulo de desenvolvimento possui a base da linguagem para comunicação de rede, construção de ambientes gráficos, suporte a internacionalização, tratamento de entrada e saída, funções matemáticas, etc.

2.6.3 Java® Enterpreise Edition (JEE)

Essa edição permite o desenvolvimento Web e aplicações corporativas. A aderência pelas corporações deu-se principalmente pela capacidade da linguagem na padronização de código e a fácil implementação de padrões de projeto. Em seu núcleo encontra-se APIs com suporte a *servlets*, suporte a JSP (*Java® Server Pages*) com um padrão para desenvolvimento de páginas internet além do suporte a containers de aplicação.

2.6.4 Java® Micro Edition (JME)

O JME atualmente está voltado para a Internet das Coisas ou desenvolvimento para dispositivos com sistemas embarcados e com capacidade desenvolvimento limitada.

Os principais componentes do JDK são:

- javac (compilador)
- java (interpretador)
- appletviewer (visualizador de applets)
- javadoc (gerador de documentação)
- jar (programa de compactação)

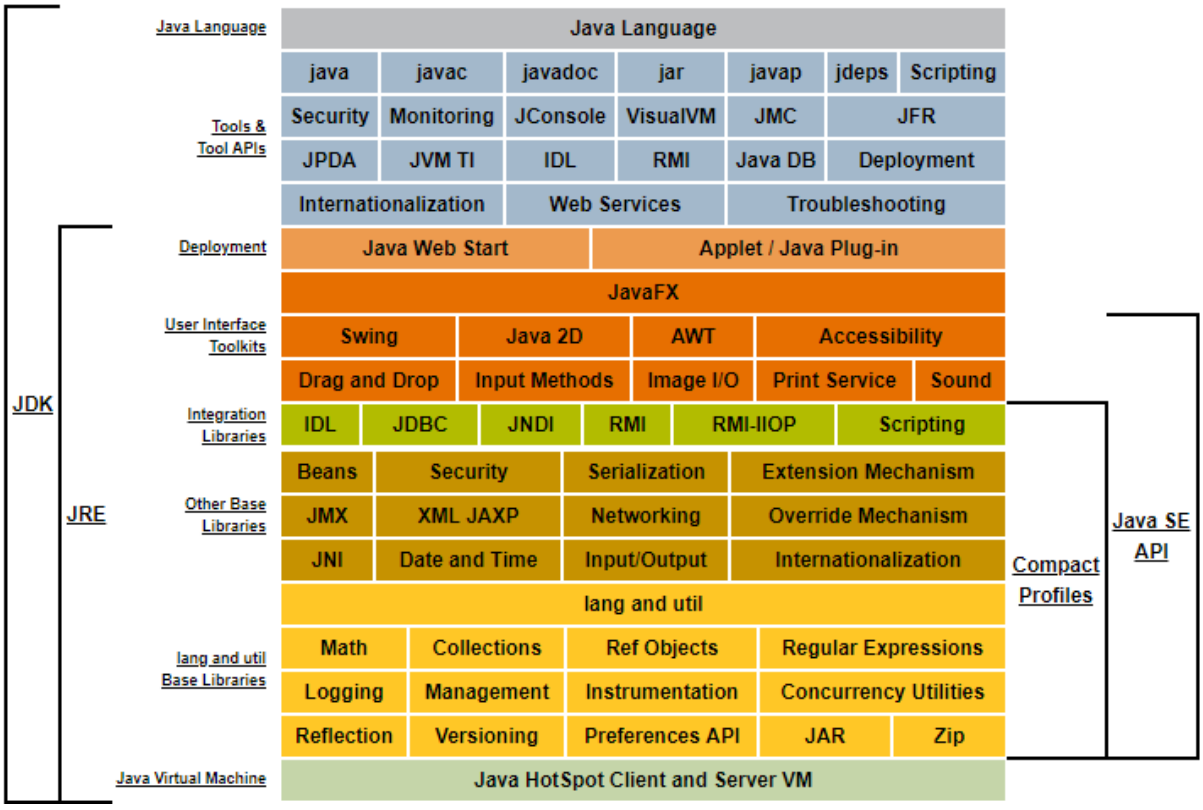


Figura 15: Arquitetura Java® versão 8.x - Fonte: The Java® Source (2019) (43).

Atualmente Java® é uma linguagem com independência de plataforma e sua arquitetura está representada na Figura 15. Para cada nova plataforma é disponibilizado uma máquina virtual capaz de traduzir os *bycodes*[4] produzidos durante a compilação.

Enquanto as linguagens mais tradicionais que utilizam compilador, necessitam de uma recompilação completa do código para que funcione em um novo hardware, o Java® necessita de apenas uma máquina virtual para rodar o código pré-compilado (Cavalcanti (2016) (42)).

2.6.5 Aspectos de arquitetura de uma Máquina Virtual Java®

A Java® Virtual Machine (JVM) pode ser definido como um computador abstrato onde todos os programas escritos na linguagem Java® são executados. Sem a camada de abstração de uma JVM não seria possível a variedade de implementações e peças de software disponíveis para as mais diversas plataformas.

Atualmente existem JVMs implementadas para computadores pessoais, sistemas operacionais, sistemas embarcados em carros, cartões de crédito, celulares, TVs, componentes de áudio, pulseiras e relógios (Vernners (2000) (44), Deitel (2016) (32)).

[4] Bytecode: instruções de máquina para uma Máquina Virtual Java (Java Virtual Machine)

Também com o aumento crescente de oportunidades de desenvolvimento de sistemas para sistemas embarcados como Internet das Coisas (IoT) e a necessidade de uma infraestrutura com alta velocidade de comunicação, tornou necessário o desenvolvimento de ferramentas capazes de rodar o mesmo código independentemente da plataforma de execução e que também atendesse de forma otimizada essa demanda.

Segundo Verners (2000) (44), para definir uma JVM é necessário, primeiramente entender três aspectos importantes durante o processo de criação e definição de uma máquina virtual desse nível

a) Aspecto de abstração da especificação
b) Aspecto da implementação da JVM
c) Aspecto de uma instancia em tempo de execução

O aspecto de abstração está relacionado com o conceito e os detalhes de implementação de uma JVM. Esse aspecto remete às especificações de construção de uma máquina virtual e o comportamento esperado de execução de um código escrito em Java®. Também estão definidos na abstração o conjunto de instruções, assim como seu significado. As instruções de máquina são chamadas de bytecodes.

A implementação concreta de uma JVM está diretamente relacionada com o código compilado e funcional de uma JVM em uma determinada plataforma. A implementação concreta é dependente do ambiente operacional em que será executado, enquanto o código escrito em Java® é independente desse ambiente. É essa característica que torna o código Java® versátil e independente de plataforma. Um arquivo binário é chamado de arquivo *class* ou *class file* sendo este independente de plataforma, ou seja, uma vez compilado rodará em qualquer JVM como suporte à aquele *class file*[5] (Deitel 2016 (32)).

Finalmente temos a instancia em tempo de execução está relacionada com o ambiente operacional ou ambiente de execução de um programa Java®. É o momento de execução de um software Java® em uma determinada plataforma, sendo que a máquina virtual irá rodar um processo independente para cada programa, criando assim uma camada de segurança na máquina hospedeira. Dessa forma, enquanto o desenvolvedor Java® se preocupa somente o

[5] Importante lembrar que apesar da maioria das documentações que tratam de programas escritos em Java® se referirem ao class file como "simplesmente" independente de plataforma, temos que notar que nem todas as máquinas vituais são capazes de rodar um codigo Java® inteiro. Como exemplo podemos citar um sistema de áudio automotivo configurado para rodar aplicações em Java®, dificilmente executará por exemplo uma aplicação Web por limitações do hardware hospedeiro.

desenvolvimento de sua aplicação o fornecedor de um determinado hardware será o responsável por fornecer a o ambiente de execução, ou Java® Runtime Environment (JRE).

A arquitetura de uma JVM possui a especificação de comportamentos no que concerne aos aspectos técnicos com subsistemas, áreas de memória, tipos de dados e instruções de máquina.

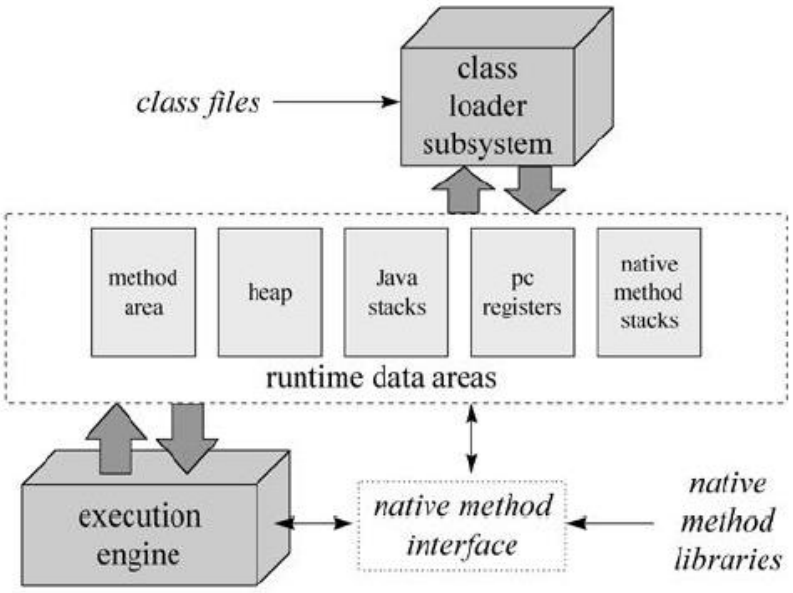


Figura 16: Arquitetura interna de uma JVM : Fonte: Verners (2000) (44)

Durante o processo de carga e execução de uma aplicação a JVM necessita de um local para armazenar informações sobre a instanciação do programa, variáveis locais, parâmetros da assinatura dos métodos, valores de retorno para cada chamada de método e valores intermediários de resultados de computação.

Apesar de cada implementação de uma JVM poder criar uma arquitetura particular com suas restrições determinadas pela plataforma, no desenvolvimento dessa JVM as linhas gerais de como deve ser o comportamento está apresentado na Figura 16. Por exemplo, algumas implementações podem tirar vantagem da memória virtual da máquina hospedeira ou usar mais memória RAM (Random Acess Memory) do que outras implementações. A natureza abstrata da especificação de uma JVM permite que seja mais fácil a implementação em uma grande variedade de computadores e dispositivos (Verners (2000) (44), Lindholm (2015) (45)).

Segundo Lindholm (2015) (45), uma JVM necessita de um *Class Loader* que é responsável por carregar os arquivos class para a máquina virtual e um motor de execução

(*execution engine*), conforme a Figura 16. Esses dois artefatos são responsáveis por distribuir as informações computadas nos blocos de:

a) Área de métodos
b) Heap
c) Pilha de primitivos Java® (Java® stacks)
d) Registradores
e) Pilha de métodos nativos

Dessa forma a JVM executa os programas e organiza a memória da máquina virtual nesses blocos que são chamados de "*runtime data areas*".

Segundo Rangel (2012) (46) o class loader não apenas é responsável por carregar dados da máquina local, mas também permite a carga a partir da rede interna ou internet. O class loader também faz a alocação e inicialização da memória e resolve as referência simbólicas e de rede.

Não é escopo desse trabalho discriminar todos os detalhes de uma JVM, entretanto na especificação definida por Lindholm (2015) (45) estão definidos diversos aspectos importantes de uma implementação de máquina virtual como os tipos de dados, comportamentos das pilhas de threads, tamanho da palavra computacional ("*word size*") e outros aspectos para o desenvolvimento de uma máquina virtual Java® completa.

Para finalizar serão descritos dois tópicos importantes de uma JVM que são o *Garbage Collector* (coletor de lixo ou de restos de objetos que estão na memória) e como é organizado a instancia de um objeto na memória de uma JVM e sua representação.

2.6.6 O Gabage Collector (coletor de lixo)

Segundo Verners (2000) (44), o termo "coletor de lixo" é uma analogia para a limpeza e consequentemente liberação de informações não uteis ao programa atual da memória do computador. O coletor possui um mecanismo autônomo de execução, isto é, não depende que um programador realize alguma ação para que o processo de limpeza inicie. Entretanto o desenvolvedor poderá sinalizar ao sistema operacional para que execute a limpeza de forma programática.

Apesar do coletor não ser um componente requerido na especificação como um processo definido que deve ser executado após um evento no sistema, a especificação apenas informa

que ele deve ser implementado e deve gerenciar a área do *Heap* de alguma maneira (Verners (2000) (44).

2.6.7 Representação de um objeto em uma JVM

A representação de um objeto em uma máquina virtual é de responsabilidade dos designers que implementam uma JVM. A especificação apenas diz que os dados primitivos devem ser representados dentro de uma instância de objeto. Esse objeto deve ter as informações necessárias de superclasses para que as regras de polimorfismo sejam respeitadas.

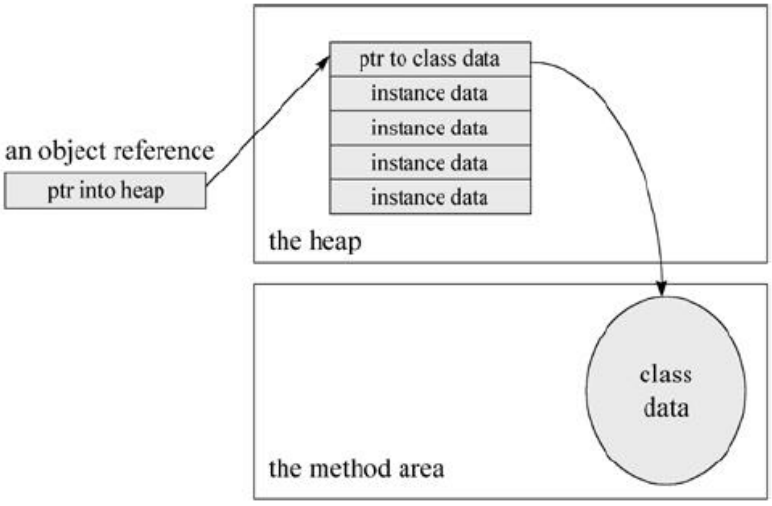


Figura 17: Representação de objeto em memória da JVM:Fonte Verners (2000) (44)

A Figura 17 representa uma forma em que um objeto instanciado é organizado na memória. Há um ponteiro para o objeto em questão que referencia os dados na área reservada de memória para o *Heap*. O mesmo o ocorre para o método que possui também uma referência, mas que é armazenado na área de métodos (Verners (2000) (44)).

Com essa configuração cada instancia de objetos tem sua própria pilha de dados, mas que compartilham a área de métodos, diminuindo assim o conteúdo da memória para toda a aplicação.

2.7 Softwares para Redes de colaboração de pesquisa

2.7.1 WEB Crawler – Extração de Dados da Web

Essa seção ir abordar os conceitos de *Web Crawler* (rastreadores web) e ao final apresentar dois grandes esforços já realizados dessas ferramentas para obtenção de dados cientométricos. O primeiro, o scriptLattes onde essa pesquisa está fortemente ligada por laços de interoperabilidade e troca de informações e o segundo é o LattesMiner, ferramenta que foi desenvolvida pelo CNPq para auxiliar instituições a obter indicadores diretamente da base da Plataforma Lattes e faz parte integrante da Plataforma Sucupira.

Outras ferramentas ainda podem ser citadas, mas está fora do escopo desse trabalho identificar e apresentar tais sistemas de software. Entretanto, durante a produção dessa pesquisa pudemos observar esforços como o Lates Extractor, SemanticLattes, GeraLattes, OntoLattes, etc.

2.7.2 Extração de dados de Pesquisadores da Web

Web Crawlers são programas criados para extrair informações de páginas da Internet. Basicamente o usuário de tais programas, informa para o programa o endereço internet da página de onde se quer buscar a informação que tipo de informação ele deve buscar (Santos (2017) (38).

Segundo Gupta (2014) (47), a extração de informação da internet exige um grande esforço por parte dos programadores de software. Existe uma grande quantidade de informação que não está disponível de forma estruturada e muitas vezes está inacessível e não pode ser obtido com um simples acesso a um hiperlink.

Entre os dados da web escondidos (*web hidden data*) (Gupta (2014) (47),) podemos citar dados estruturados em banco de dados como um catálogo de produtos, catálogos de bibliotecas, imagens de satélite.

Outra forma de informação na web oculta é aquela que são apresentados após o preenchimento de um formulário web. Muitos desses dados são resultados dinâmicos ou de tempo real. Um exemplo clássico de dado de tempo real são promoções de passagens aéreas onde a informação fica disponível por pouco tempo, portanto um acesso realizado mais de uma vez poderá apresentar resultados diferentes após o primeiro acesso (Gupta (2014) (47)).

Entretanto com a explosão da Web, a quantidade de informação útil, relevante e pronta para consumo pelas empresas e instituições governamentais tiveram um crescimento exponencial nos últimos anos.

Conforme Reis (2013) (48), a estrutura preferida em que a Web está organizada, está no formato HiperText Markup Language (HTML). Dessa forma o código fonte de uma página Web possui duas informações importantes: os hiperlinks que conectam uma página Web a outra página e a estrutura de formatação de layout.

Código fonte é o formato sintático de uma linguagem de programação. No caso da Web o HTML não é uma linguagem de programação, mas uma linguagem de marcação ou de layout baseada em XML. De qualquer forma ela apresenta uma estrutura lógica para que navegadores Web possam *renderizar*[6] uma informação na tela do computador.

Essa natureza das páginas HTML permite a um software *crawler* (rastreador) analisar o conteúdo de uma página web sabendo para onde seguir - passar para uma próxima página – usando os hiperlinks e recuperar dados a partir da formação dos campos.

Ainda conforme explica Reis (2013) (48), uma execução clássica de um rastreador web é obter os dados de uma página web denominada semente, analisar e obter as informações relevantes da página a partir de seu código fonte, verificar os dados, observar se há novas páginas para visitar (através dos hiperlinks encontrados). Caso tenha sucesso o software pode decidir se deve parar, uma vez que alcançou as regras necessárias predefinidas e finalizar o processamento ou continuar executando os passos anteriores em um laço de repetição. A Figura 18 demonstra esse procedimento sugerido.

[6] Renderizar: apresentar dados na tela em forma específico para atender a uma determinada regra para visualização.

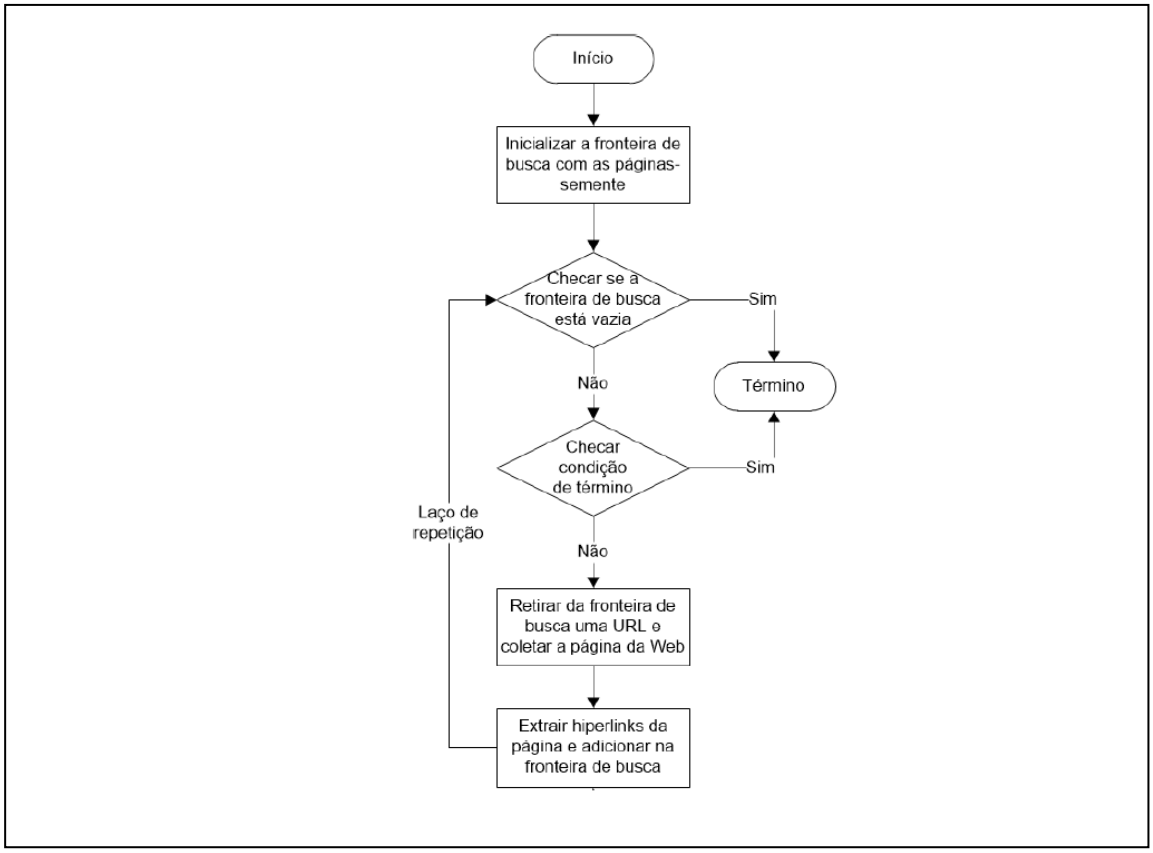


Figura 18: Análise e coleta de dados de um Web Crawler: Fonte Reis (2013) (48)

2.7.3 scriptLattes

Em 2014 a ferramenta scriptLattes, recebeu uma atualização para gerar informações em formato XML (eXtensible Markup Language) de todos os currículos que essa ferramenta processa. Pesquisadores e instituições dedicam enormes esforços para desenvolver ferramentas de software capazes de gerencias objetos digitais.

O scriptLattes tem duas versões. A primeira foi escrita em Perl mas o projeto evoluiu e foi totalmente reescrito em Python. A versão utilizada nesse trabalho é a versão Python

Tipicamente executa-se o scriptLattes utilizando-se o sistema operacional Linux. Apesar de não haver barreira em utiliza-lo em ambientes Windows ou MacOS, o sitio que hospeda a ferramenta somente fornece informações de como instalar as dependências em Linux. Existem uma série de bibliotecas necessárias que são utilizadas durante a execução do programa e para instala-las executa-se os seguintes comandos:

- sudo apt-get install python-all python-setuptools python-utidylib python-matplotlib python-levenshtein python-pygraphviz
- sudo apt-get install python-numpy tidy python-scipy python-imaging python-mechanize python-pandas
- sudo easy_install pytidylib

Após a instalação das dependências acima, o programa deverá ser executado de forma fácil.

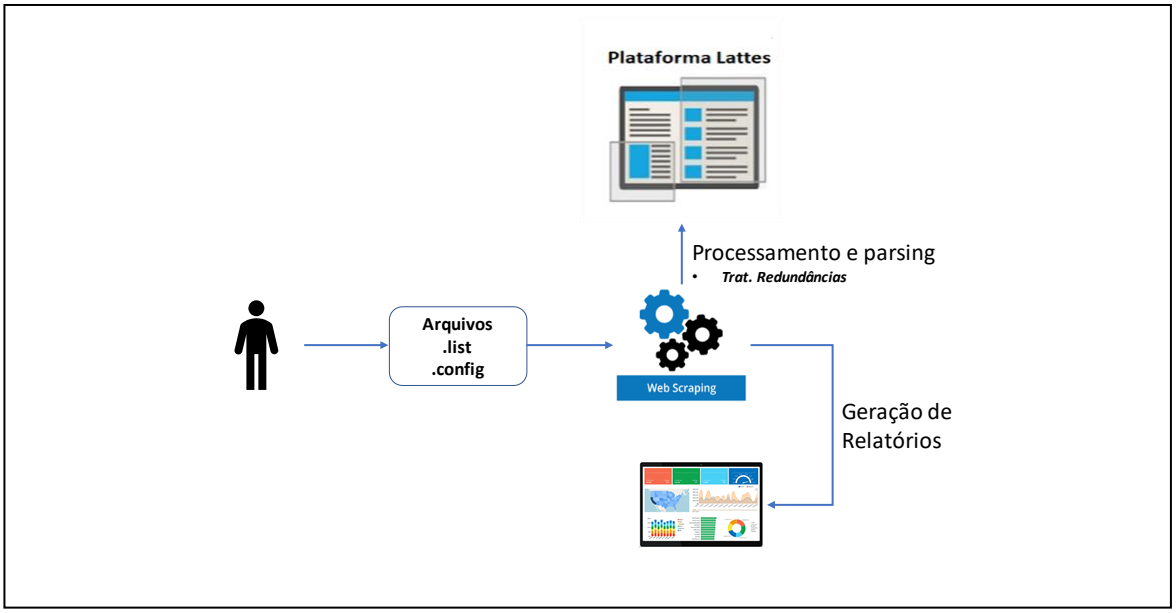


Figura 19: Web Crawler scriptLattes (processo de extração resumido)

O esquema de execução do scriptLattes é realizado conforme a Figura 19. Dois arquivos principais são utilizados

- Arquivo com extensão “.list”
- Arquivo com extensão “.config”

As extensões são meramente arbitrárias, entretanto por boas práticas sugerimos que se mantenha o padrão.

O arquivo “. list” contém a lista de pesquisadores e possui o seguinte layout

<nome >, <id_lattes_16 >, <periodo_de_extracao> , <codigo>

- **nome**: Nome do pesquisador no qual está se buscando o dados do Lattes
- **id_lattes_16**: Codigo de 16 digitos que identifica o pesquisador na base Lattes
- **periodo_de_extracao**: Faixa de anos em que será extraído os dados

- **codigo**: Uma cadeia de caracteres qualquer que será devolvida após a extração

Como dados de entrada foi utilizado os dados de orientadores descritos na seção anterior e geração do arquivo “unifesp_<cod_capes_programa>.list”

Após a preparação do arquivo “.list” o próximo passo é alterar arquivo “.config”. Nesse arquivo deve ser ajustado basicamente o local de onde o programa irá encontrar o arquivo “.list”. Há uma série de outras configurações bem documentadas, mas para o escopo desse trabalho foi deixado o restante das configurações no padrão.

A execução do scriptLattes é através do terminal de comando do Linux é realizada pelo seguinte comando:

- python ./scriptLattes <caminho_arquivo_config.list>

2.7.4 LattesMiner

O Lattes Miner é uma ferramenta escrita em Java® de acesso restrito às instituições e desenvolvida pelo CNPq como uma ferramenta para obtenção de informações de pesquisadores das instituições licenciadas (Alves (2012) (49)).

A proposta do Lattes Miner é permitir que outros desenvolvedores implementem suas próprias rotinas de desenvolvimento de alto nível e que atendam às necessidades específicas a partir da integração das classes Java® no projeto particular.

Conforme Santos (2017) (38), uma característica importante do LattesMiner é a capacidade dessa ferramenta obter as informações do ID do Lattes, facilitando o trabalho de recuperação das informações do Pesquisador por outros crawlers web. Um ID Lattes é um identificar único de um Pesquisador dentro da base da Plataforma Lattes

Para obtenção desses IDs o desenvolvedor utiliza uma das interfaces de programação oferecida pela ferramenta fornecendo como arquivo de entrada a lista de nomes de pesquisadores e um caminho de saída para a lista de IDs encontados(Alves (2012) (49)).

A Figura 20 demonstra o código Java® necessário e os arquivos de entrada e saída respectivamente.

```
import java.util.*;
import lattes.util.Util;
import static lattes.miner.LattesMiner.*;

public class Exemplo
{
 public static void main(String[] args)
 {
  List<String> list = new ArrayList<String>();

  for (String nome : Util.getList("nomes.txt"))
   list.add(search(nome));

  Util.setList(list, "ids.txt");
 }
}
```

ENTRADA [nomes.txt]

```
Nelson Maculan Filho
Luiz Bevilacqua
Fernando Galembeck
Alvaro Toubes Prata
João Fernando Gomes de Oliveira
```

SAÍDA [ids.txt]

```
K4783153E3
K4787137U2
K4787937A7
K478159Z8
K4787011P6
```

Figura 20: Exemplo de execução LattesMiner. Fonte: Santos (2017) (38)

Conforme cita Alves (2012) (49):

> *"A linguagem LattesMiner faz parte de um projeto maior denominado "Sistema Unificado de Currículos e Programas: Identificação de Redes Acadêmicas – SUCUPIRA". O projeto SUCUPIRA ... visa ser uma ferramenta computacional automatizada e de domínio público que possa eventualmente auxiliar na obtenção de indicadores de desempenho de docentes, pesquisadores, e programas de pós-graduação"*

Um aspecto negativo dessa ferramenta é que a extração das informações só pode ser executada através de instituições credenciadas e com acesso às informações de pesquisadores e docentes da institucionais. Portanto ela não permite fazer análises mais complexas e geração de indicares que façam comparativos com outras universidades ou analisar o comportamento cientométrico fora da instituição(Santos (2017) (38)).

3 OBJETIVOS

Desenvolver uma solução tecnológica para análise da produção científica de uma Rede de colaboração de pesquisadores a partir de dados extraídos da Plataforma Lattes.

3.1 Objetivo secundário:

Para que ser possível alcançar o objetivo proposto, será necessário desenvolver dois passos complentares:

1) Descrever uma arquitetura de solução tecnológica de análise quantitativa da produção científica de uma Rede de colaboração de pesquisadores a partir de dados extraídos da Plataforma Lattes;

2) Apresentar um estudo de caso a partir dos resultados obtidos pela solução tecnológica desenvolvida para análise de uma Rede de colaboração de pesquisadores de certa área do conhecimento de uma universidade pública federal localizada no estado de São Paulo.

4 JUSTIFICATIVA

A peça de software proposta tira proveito de outras ferramentas para que não fosse preciso “reinventar a roda” e explora outros formatos de análise, expandindo seu potencial computacional.

Como benefício proposto, será apresentada uma rede de colaboração real de pesquisadores para demonstrar as potencialidades que podem ser obtidas.

A partir do software desenvolvido, o mesmo poderá servir como um modelo para criação de novos softwares e derivações dos atuais existentes para pesquisadores interessados em análise cientométrica.

Ademais, o estudo de caso escolhido é aderente ao Programa de Mestrado Profissional em Tecnologia, Gestão e Saúde Ocular, que se encontra dentro da Área de Avaliação da Medicina III da CAPES e no Departamento de Oftalmologia da UNIFEP.

5 ESTRUTURA DO TRABALHO

5.1 Problema de pesquisa

Como extrair informação consistente, compilada e de valor prático através de uma solução tecnológica de análise cientométrica, ou seja, aquela que baseia seus dados nas informações e meta-informações cientificas aplicada a um cenário extração de informação da Plataforma Lattes e as Redes de Colaboração?

5.2 Dados do objeto escolhido

A Rede de pesquisadores definida para o caso está relacionada apenas aos docentes permanentes ligado aos programas de pós-graduação stricto sensu da Universidade Federal de São Paulo - UNIFESP e vinculados à Área de Avaliação da Medicina III da CAPES.

Estudo transversal, com dados referentes a Produção bibliográfica de artigos completos publicados em periódicos, dentro do período de 2010 a 2018, sem realização de análise da evolução da rede, ano a ano.

Foram considerados os docentes permanentes registrados na Plataforma Sucupira e na base de dados registrada na UNIFESP com ao menos um vínculo em cada ano dentro do período entre 2010 a 2018. Os dados encontrados na Plataforma Sucupira foram mesclados com os dados da UNIFESP e verificado se haviam divergências quantitativas e de pessoas registradas. Foram excluídos desta pesquisa docentes visitantes ou colaboradores, além de discentes, egressos e demais pesquisadores de outras instituições. São no total 10 programas, que totalizam 10 cursos analisados.

6 MÉTODOS E INSTRUMENTOS

A extração de informações em formato textual é uma necessidade emergente na atualidade. Com a grande quantidade de informações disponibilizadas na Internet ficou evidente que seria necessário a criação de ferramentas computacionais e peças de software que tivessem como prerrogativa a extração de dados de páginas da Internet. Dessa forma, os dados de interesse serão extraídos do sitio de Internet da Plataforma Lattes.

O modelo de extração, processamento e aplicação de regras segue o diagrama da Figura 21 que será explicitado com mais detalhes adiante.

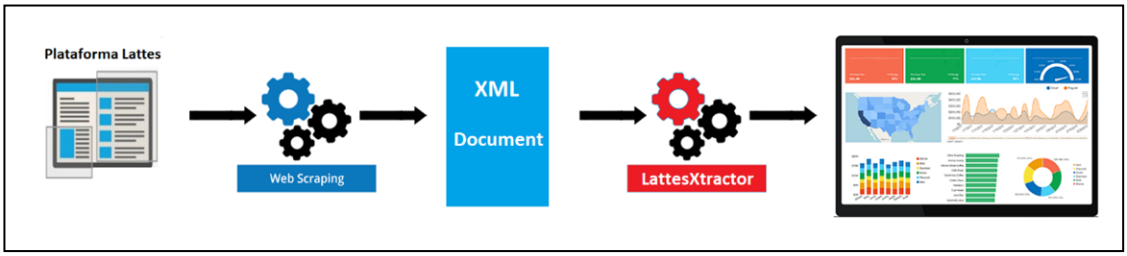


Figura 21: Processo de Extração de dados. Fonte: próprio autor. São Paulo, SP. 2019

Conforme a Figura 21, a Plataforma Lattes possui um repositório de dados rico em informação de pesquisadores e da pesquisa brasileira. Utilizando técnicas de *Web Scrapping*[7] juntamente com ferramentas de *web crawlers*[8] é gerado um arquivo XML como resultado do processamento. Esse XML é um arquivo estruturado e de entrada para a ferramenta proposta nessa pesquisa, denominada *LattesXtractor*. Por sua vez o *LattesXtractor* irá processar a informação, organizar os dados em memória e disponibilizar para os desenvolvedores uma interface para acesso e manipulação dos dados. Com a informação processada é possível realizar análises quantitativas e qualitativas dos dados do Lattes.

6.1 Delimitação do estudo

Trata-se de um estudo descritivo de natureza exploratória com uma abordagem de inovação no campo em que se situa, cujo objetivo foi desenvolver uma aplicação com linguagem de programação Java®.

[7] *Web Scrapping*: Técnica para extrair de forma automática utilizando robôs (software programado) que lêem páginas na internet e retornam informação.

[8] *Web Crawlers*: Robôs programados para ler dados na Internet

6.2 Classificação do trabalho de pesquisa

A classificação dessa pesquisa está delineada em aspectos organizados em tópicos seguindo a seguinte ordem:

6.2.1 Abordagem e organização

A escolha da abordagem foi definida como de caráter qualitativo, sem vinculação estatística. Entretanto os resultados também são avaliados em uma abordagem secundária sob a perspectiva quantitativa, permitindo uma mensuração dos resultados alcançados através da análise dos gráficos de colaboração como um resultado indireto e de efeito colateral dos resultados obtidos.

6.2.2 Natureza

Desenvolvimento de software. A natureza desse trabalho é realizar uma pesquisa aplicada tendo como resultado o desenvolvimento de uma aplicação escrita em linguagem de programação Java®. O foco da aplicação e a extração de dados ofertados pela Plataforma Lattes através da leitura de um arquivo XML.

6.2.3 Procedimentos para realização das pesquisa e aspectos técnico

Os procedimentos para obter os resultados estão definidos em três aspectos que dão forma ao projeto:

1. Pesquisa bibliográfica com referencial teórico sob o tema tratado nessa pesquisa.
    a. Engenharia de Software aplicada no processo de desenvolvimento
    b. Conceitos de Redes de Colaboração (RC) e Analise de Redes Sociais (ARS) para dar sustentação ao estudo de caso
2. Desenvolvimento de uma aplicação, escrita em linguagem Java® que atenda aos critérios e objetivos da pesquisa;
3. Apresentação de um estudo de caso para demonstrar o potencial da ferramenta;

6.3 Software utilizado e número de versão

Abaixo estão relacionados software e equipamentos utilizados durante o desenvolvimento dessa pesquisa e os locais de onde foram obtidos. Durante o processo, outras versões mais antigas de software e hardware foram utilizadas, entretanto no estado da arte dessa pesquisa as versões mais recentes utilizadas estão listadas abaixo:

6.3.1 Software

1. Netbeans versão 8.2 – 64 bits
    a. IDE de Programação
2. Java® JDK – Kit de desenvolvimento
    a. Java® version "1.8.0_201"
    b. Java®(TM) SE Runtime Environment (build 1.8.0_201-b09)
    c. Java® HotSpot(TM) 64-Bit Server VM (build 25.201-b09, mixed mode)
3. Xtream – Framewok para leitura de arquivos XML
4. scriptLates
    a. Versão 8.0 com motor de leitura de captcha[9]
5. Gephi versão 0.9.2 – Visualizador de grafos

6.3.2 Hardware

1. Notebook Sony Vayo
    a. Processador Intel i7
    b. HD 1Tb
    c. 8 Gb memória RAM

6.4 Análise e descrição e dos processos

6.4.1 Construção do Software

Para essa pesquisa foi desenvolvido uma peça de software capaz de obter os dados da Plataforma Lattes a partir de um arquivo XML que contém dados de um grupo de pesquisadores. Após o processamento do XML foi organizado em memória uma estrutura de dados robusta permitindo que as informações assim dispostas poderão ser processadas de forma a atender necessidades especificas de análises.

6.4.2 Arquitetura do LattesXtractor

O software LattesXtractor foi desenvolvido utilizando linguagem Java®, de forma modular e orientado a objetos.

[9] A Plataforma Lattes tem adotado a medida de incluir um captcha (imagem com números e letras) para acesso aos currículos o que tem inviabilizado muitas pesquisas que necessitem capturar as informações dos dados de pesquisa por ferramentas de software. O scriptLattes tem implementado um sistema de OCR (Optical Character Recognition) para leitura da imagem de forma automática. Maiores informações podem ser obtidas no site oficial do scriptLattes.

A organização de objetos foi necessária para abstrair a informação do currículo lattes. A Figura 22 mostra na forma de diagrama de classes a forma como a informação está organizada[10].

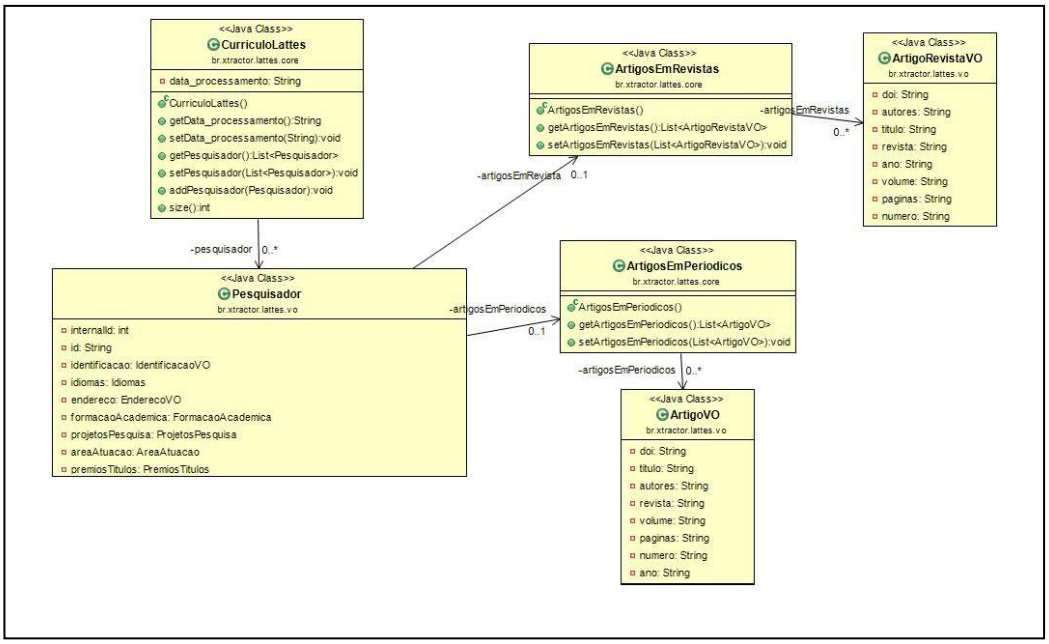


Figura 22: Diagrama de classes simplificado. Fonte: próprio autor. São Paulo, SP. 2019

A classe *CurriculoLattes* é uma abstração do currículo lattes e contém uma lista de pesquisadores. Possui uma interface bem definida por onde pode-se buscar as informações necessárias através de métodos de chamadas de sistema. Para cada chamada de sistema, a ferramenta devolve uma outra lista referente ao que foi solicitado.

Assim temos um conjunto de pesquisadores em uma lista e para cada pesquisador da lista obtém-se listas de informações referente à aquele pesquisador.

Por exemplo, para um pesquisador obtém-se uma lista artigos em periódicos que esse pesquisador publicou. Para obter o artigo em questão, basta realizar uma chamada de sistema tal que ele recupere um objeto chamado *ArtigoVO*. O mesmo se dará para outros objetos como *ArtigoEmRevistaVO*. Primeiro obtém-se o pesquisador, em seguida obtém-se a lista de *ArtigosEmRevistas* e por último o dado em si que está armazenado no objeto *ArtigoRevistaVO*[11].

[10] A Figura 22 apresenta somente algumas classes que compõe a ferramentas LattesXtractor. Não seria possível colocar todas as classes organizadas por limitação do espaço físico da página.

[11] O sufixo VO é um acrônimo que se refere a Value Object, ou objetos que contém conteúdo em si. Na literatura esse tipo de objeto também pode ser encontrado com o nome de POJO (Plain Old Java® Object) ou Java®Beans. O POJO é mais utilizado para classes que não dependem de herança ou não existe uma relação

A Figura 23 mostra o diagrama de classes acima em uma estrutura de árvore para melhor visualização.

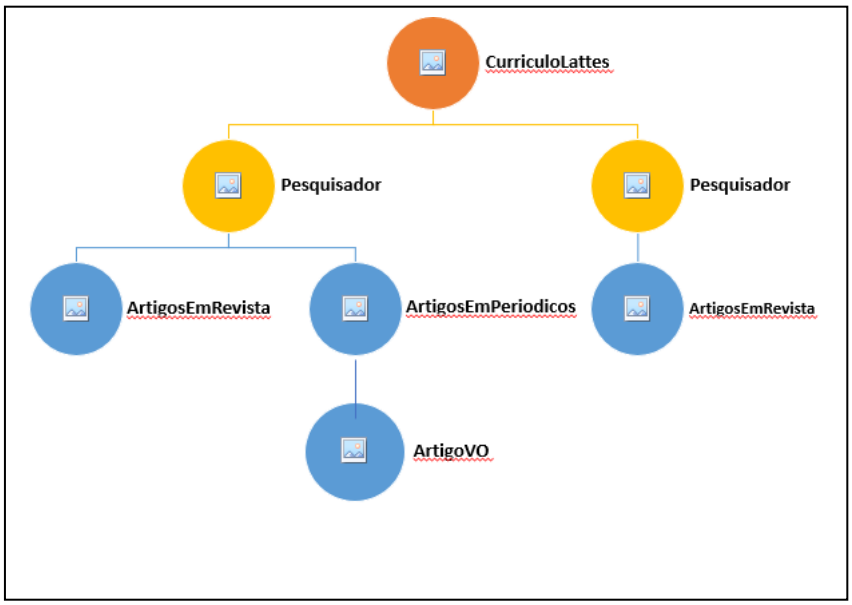


Figura 23: Extrutura de classes em uma visualização em árvore. Fonte: próprio autor. São Paulo, SP. 2019

Observa-se na Figura 23 que a classe *CurriculoLattes* é o vértice raiz da árvore e possui uma série de nos filhos aqui definidos pela classe *Pesquisador*. Cada elemento da classe *Pesquisador* tem seu conjunto de listas e de informações referentes e específicas de um determinado pesquisador.

A estrutura é montada a partir da leitura de um arquivo XML que é criado após a execução do scriptLattes e que será detalhada mais adiante.

A classe *LattesXtractorGUI* é o ponto de entrada da ferramenta. Conforme o digrama de classes Figura 24, observamos que ela contém o motor de leitura de dados do XML representado pela classe *ScriptLattesXmlParser*.

programática com frameworks. Com a evolução do Java®, objetos do tipo VO passaram a ter uma dependência com a classe Serializable para transferência em rede. Essa característica juntamente com outras, permitiram a evolução do nome para *Java®Beans*.

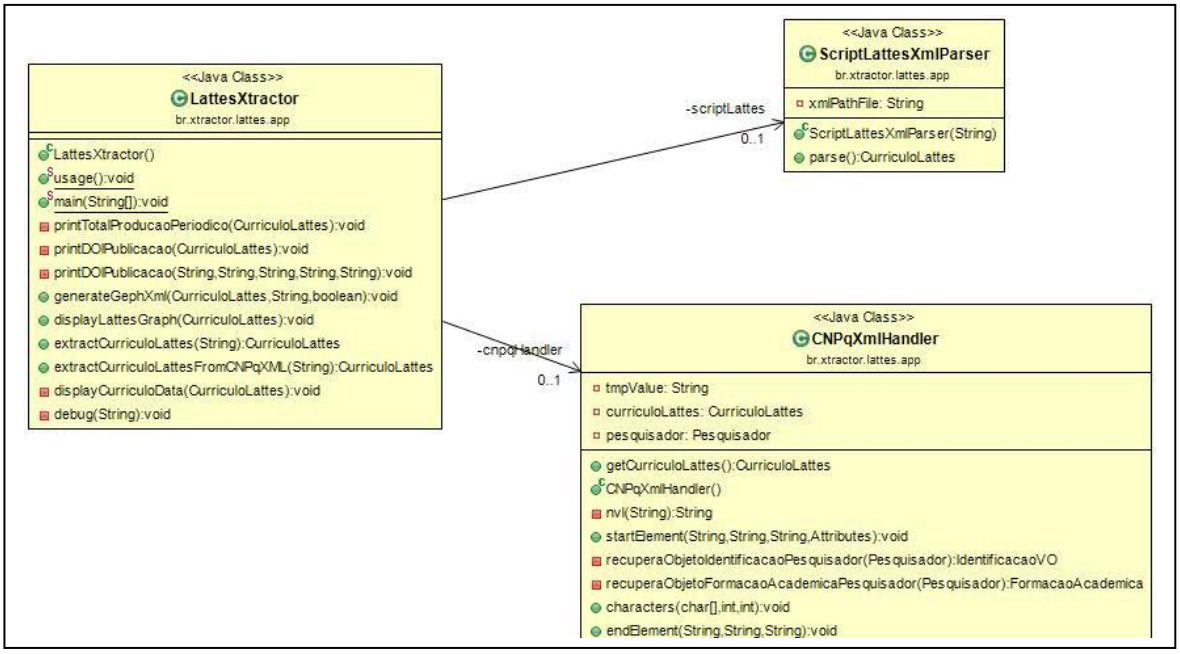


Figura 24: Núcleo do LattesXtractor. Fonte: próprio autor. São Paulo, SP. 2019

O objeto *ScriptLattesParser* tem todas as regras de leitura de um arquivo XML disponibilizado pelo *scriptLattes* escrito em Python. Uma vez executado o *LattesXtractor*, este irá instanciar a classe *ScriptLattesParser* que ao final de seu processamento irá retornar uma instancia da classe *CurriculoLattes* com a lista de todos os pesquisadores obtidos no XML juntamente com as informações de cada pesquisador.

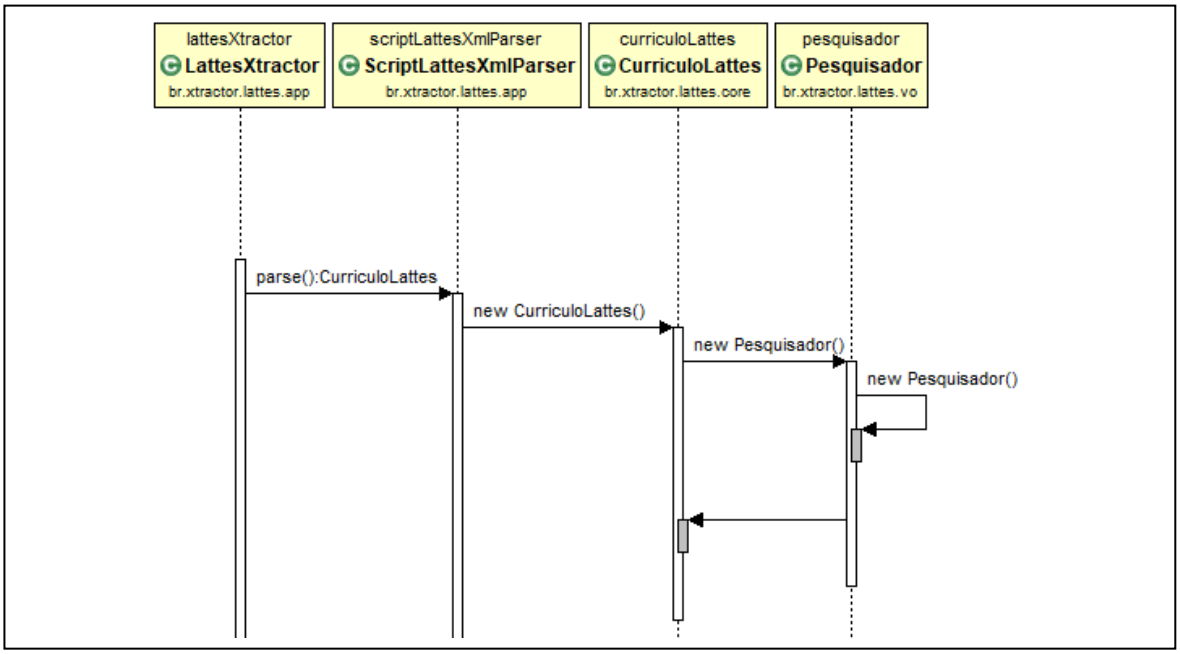


Figura 25: Diagrama de sequência simplificado do processo de leitura do XML. Fonte: próprio autor. São Paulo, SP. 2019

O diagrama de sequência da Figura 25 apresenta de forma simplificada o fluxo de execução da ferramenta até a obtenção dos dados dos pesquisadores. Os objetos instanciam uns aos outros através da troca de mensagens até obter todas a informações. Todo o processamento de criação de classes e organização em memória é realizado automaticamente. O capítulo 2.6 Linguagem Java® detalha o mecanismo de troca de mensagens e instanciação de objetos.

6.4.3 Interoperabilidadeda da ferramenta LattesXtractor com o scriptLattes

A ferramenta LattesXtractor, objeto dessa pesquisa, integrou-se de forma parcial com a ferramenta de *Web Crawler* "*scriptLattes"* com o objetivo de alavancar a força do software já existente propondo a interoperabilidade entre as ferramentas.

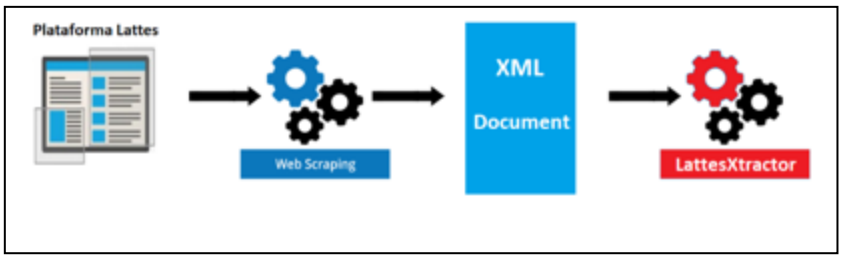


Figura 26: Integração do LattesXtractor com o scriptLattes. Fonte: próprio autor. São Paulo, SP. 2019

Por se tratarem de programas diferentes, inicialmente executa-se o scriptLattes com todos os parâmetros e informações necessárias (*para maiores informações veja o capítulo Web Crawler*). Ao terminar o processamento o scriptLattes irá salvar o arquivo database.xml no diretório de execução da ferramenta. Por sua vez executa-se o *LatteXtractor* informando o nome o caminho do arquivo gerado pelo scriptLattes.

Para execução do *LattesXtractor*, executa-se a classe *LattesXtractorGui*. Essa classe abrira uma interface gráfica para manipulação do arquivo XML

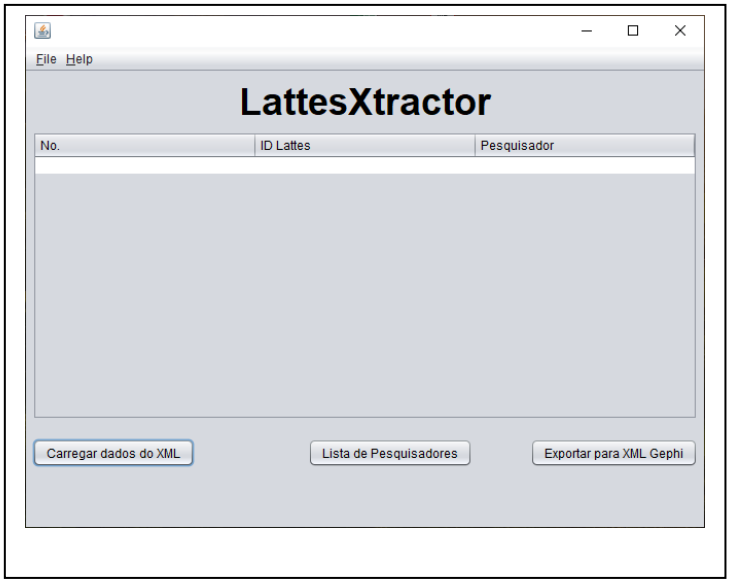


Figura 27: LattesXtracotorGUI - Interface gráfica para o LattesXtractor

Ao clicar no botão “Carregar dados do XML” abrirá uma janela para selecionar o arquivo database.xml (ou outro nome que tenha sido gerado pelo motor do scriptLattes). Caso o arquivo XML seja um arquivo inválido, uma mensagem de erro será mostrada para o usuário. Se a leitura do XML ocorrer como esperado a janela da Figura 28 será apresentada com a lista de pesquisadores lidos.

Nessa tela será apresentado o ID Lattes do Pesquisador que é um identificador único de identifica de forma inequívoca o pesquisador dentro da Plataforma Lattes e o nome completo lido na base.

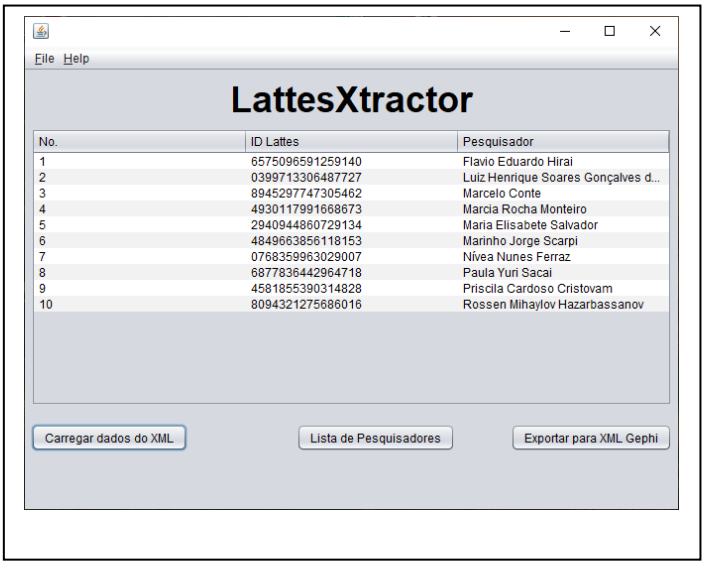


Figura 28: Pesquisadores carregados com sucesso na ferramenta

Uma vez com os dados carregados é possível ainda visualizar dados carregados dos pesquisadores clicando no botão “Lista de Pesquisadores”. Uma nova janela se abrirá e será possível navegar entre os pesquisadores lidos.

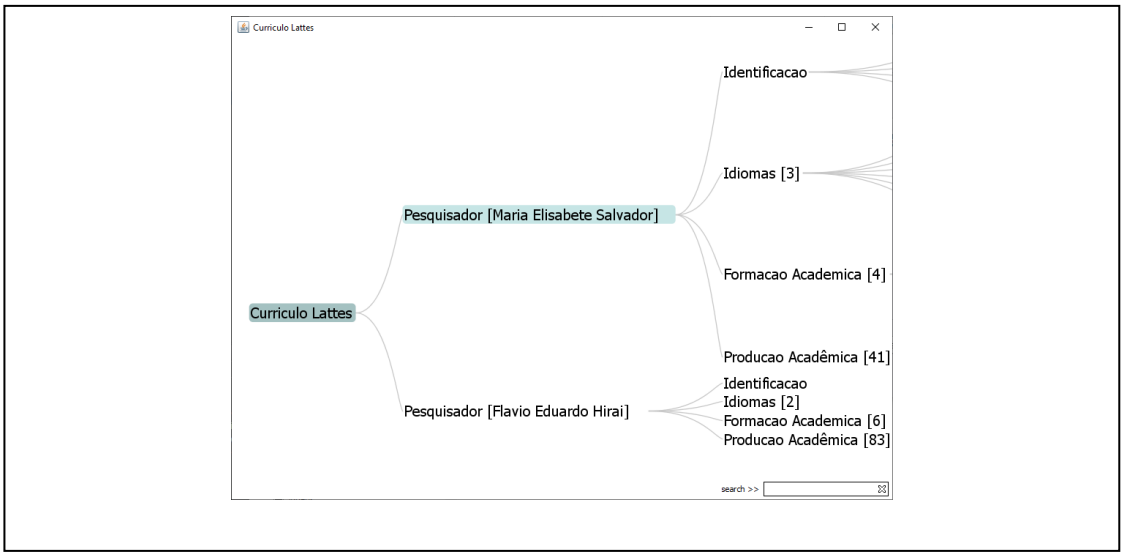


Figura 29: Visualização das informações dos pesquisadores

É possível visualizar informações importantes sobre as pesquisas realizadas, como a produção acadêmica, além da identificação, formação acadêmica do pesquisador. Essa tela é importante pois é possível verificar e validar os dados coletados da Plataforma Lattes. Caso tenha algum erro, basta o pesquisador atualizar na Plataforma Lattes e rodar novamente o programa.

Por fim, o botão “Exportar para XML Gephi”, a ferramenta irá fazer a exportação de relacionamento de grafo em um arquivo XML para ser lido pelo software Gephi. No

Anexo 4 | Aplicação da rotina no Gephi , página 119 está descrito todo o processo leitura e transformação desse dado.

6.4.4 Obtenção e organização das informações de pesquisadores

O universo de dados utilizado foi disponibilizado pelo Departamento de Tecnologia de Informação (DTI) da Universidade Federal de São Paulo (UNIFESP) com o nome dos pesquisadores e o ID Lattes. O ID Lattes é um número de 16 dígito que identifica um pesquisador de forma única dentro da plataforma Lattes e qualquer pessoa que crie um currículo nessa plataforma recebe esse número.

Para a seleção dos dados, o critério de organização dos dados para obtenção da base inicial foi que os pesquisadores deveriam ser docentes do *Stricto Sensu* e serem orientadores credenciados na data da extração, isto é, serem orientadores de alunos de níveis mestrado e/ou doutorado nos programas de Pós-Graduação oferecidos pela UNIFESP da área de Medicina III.

Entretanto entendendo que esse universo de dados é um universo de fluxo continuo, isto é, docentes iniciam atividades de orientação e se desvinculam dos programas com frequência, foi realizado uma segunda busca de informações na Plataforma Sucupira para obter dados oficiais de orientadores cadastrados.

Essa segunda busca, entretanto, traz uma desvantagem: a base de extraída possui nome do programa e o nome do pesquisador sem o ID Lattes.

Para contornar esse problema colocou-se as duas bases em uma planilha Excel e realizou-se uma busca que coincidisse o nome do pesquisador nas duas bases. Caso encontra-se a semelhança, aplicava-se na célula logo após o nome do pesquisador o ID Lattes obtido da base da UNIFESP. Para os nomes em que não foi encontrado correspondência, obteve-se o ID Lattes foi realizado uma busca na Plataforma Lattes baseado no nome do pesquisador.

Dessa forma foi possível criar uma base de dados final que serviu como fonte para extração dos dados disponibilizados na Plataforma Lattes. A Tabela 1 mostra um exemplo dos dados obtidos.

ID Lattes	Pesquisador
3089430786971948	A.L.B.L.D.B.
4595643076722509	M.G.R.D.G.
.....	.....
1311802831007681	W.V.V.

Tabela 1: Exemplo de lista de orientadores credenciados na UNIFESP. Fonte: próprio autor. São Paulo, SP. 2019

Como a base de dados é uma base real, os nomes dos pesquisadores foram reduzidos às suas iniciais, para atender a normativas de privacidade.

6.4.5 Organização de informações e criação de grupos

Para a obtenção e organização de informações, a extração dos dados de orientadores da UNIFESP, tomou-se o cuidado de se obter os seguintes campos para compor as informações:

- Código CAPES do Programa
- Nome do Programa
- Nome do Orientador
- Código Lattes de16 dígitos

A Tabela 2 ilustra o resultado dessa extração.

	CPF	COD_CAPES	PROGRAMA	NOME_ORIENTADOR	COD_LATTES_10	COD_LATTES_16
2	00450079341	33009015088P9	Alimentos, Nutrição e Saúde	ANNA RAFAELA CAVALCANTE BRAGA	K4713925D2	1145965338858435
3	35258199860	33009015088P9	Alimentos, Nutrição e Saúde	BARBARA PEREZ VOGT	ID=K420211	
4	28626828888	33009015088P9	Alimentos, Nutrição e Saúde	CAMILA APARECIDA MACHADO DE OLIVEIRA	K4734809T4	4886067148875464
5	83054227472	33009015088P9	Alimentos, Nutrição e Saúde	CRISTIANO MENDES DA SILVA	K4767813U6	7868915353525184
6	26786337805	33009015088P9	Alimentos, Nutrição e Saúde	DANIEL ARAKI RIBEIRO	K4707522T6	9969803499258672
7	29422441862	33009015088P9	Alimentos, Nutrição e Saúde	DANIEL HENRIQUE BANDONI	K4757536U5	6104429791974852
8	29187299810	33009015088P9	Alimentos, Nutrição e Saúde	DANIELLE ARISA CARANTI	K4759648D5	4760019839583649
9	15128438813	33009015088P9	Alimentos, Nutrição e Saúde	ELKE STEDEFELDT	K4772727P6	5590674723055512
10	94782695349	33009015088P9	Alimentos, Nutrição e Saúde	ITALO BRAGA DE CASTRO	K4705887T3	0426330018731301
11	52968421620	33009015088P9	Alimentos, Nutrição e Saúde	JUAREZ PEREIRA FURTADO	K4787946J6	6869345414404363
12	26079150824	33009015088P9	Alimentos, Nutrição e Saúde	LUCIANA PELLEGRINI PISANI	K4742895D8	3983527783636073
13	12069390870	33009015088P9	Alimentos, Nutrição e Saúde	MARCOS HIKARI TOYAMA	K4763204P3	8573195327542061
14	20307659453	33009015088P9	Alimentos, Nutrição e Saúde	MARIA ANGELICA TAVARES DE MEDEIROS	K4785184D8	4891875284385301
15	01544374097	33009015088P9	Alimentos, Nutrição e Saúde	MARIA LAURA DA COSTA LOUZADA	K4215088H1	4542068707177097
16	25061170830	33009015088P9	Alimentos, Nutrição e Saúde	ODAIR AGUIAR JUNIOR	K4795500A8	0398348332863521
17	05839055875	33009015088P9	Alimentos, Nutrição e Saúde	PATRICIA DA GRACA LEITE SPERIDIAO	K4763857P6	7520873457028761
18	25837365826	33009015088P9	Alimentos, Nutrição e Saúde	PAULA ANDREA MARTINS	K4775858P7	1364300323959453
19	15004213836	33009015088P9	Alimentos, Nutrição e Saúde	SEMIRAMIS MARTINS ALVARES DOMENE	K4790280D3	7373562130327980
20	30379157810	33009015088P9	Alimentos, Nutrição e Saúde	VANESSA DIAS CAPRILES	K4733346D3	2781360640415360
21	02306871990	33009015088P9	Alimentos, Nutrição e Saúde	VERIDIANA VERA DE ROSSO	K4762154U7	4938721558237749
22	42173566091	33009015087P2	Análise Ambiental Integrada	ANA LUISA VIETTI BITENCOURT	K4785982Y0	1230773059940967
23	10210054875	33009015087P2	Análise Ambiental Integrada	ANDREA RABINOVICI	K4773629J6	4506171831521594
24	26112502860	33009015087P2	Análise Ambiental Integrada	CAMILO DIAS SEABRA PEREIRA	K4779002E2	4298143669596994
25	00751918814	33009015087P2	Análise Ambiental Integrada	CLAUDIO BENEDITO BAPTISTA LEITE	K4709269Y5	5991628846150178
26	15992167870	33009015087P2	Análise Ambiental Integrada	CRISTINA ROSSI NAKAYAMA	K4792501P0	5087991903554322

Tabela 2: Exemplo de dados extraídos da base institucional da UNIFESP

Dessa forma foi possível categorizar orientadores por grupos onde o Programa é a chave para o corte, através de uma query SQL (Strutured Query Language) aplicada na base de dados

Oracle da UNIFESP. O resultado é um arquivo texto com extenção CSV, facilmente lido pelo Excel e apresentado na Tabela 2.

```
SELECT DISTINCT
  APRIN.COD_CAPES
  , APRIN.DESCRICAO PROGRAMA
  , APRIN.CODIGO
  , VINCS.NOME || ' ' || VINCS.SOBRENOME NOME_ORIENTADOR
  , PGORI.COD_LATTES_10
  , PGORI.COD_LATTES_16
  , NIV.DESCRICAO NIVEL_ORIENTACAO
  , SITCRED.DESCRICAO
  , CREDORI.SIT_CRED_ID
FROM
  ACADEMICO.VINCS VINCS
  JOIN ACADEMICO.PG_ORIENTADORES PGORI
    ON VINCS.TIPO = PGORI.TIPO
    AND VINCS.NUMERO = PGORI.NUMERO

  JOIN ACADEMICO.PG_CRED_ORIEN CREDORI
    ON CREDORI.ORIEN_ID = PGORI.ID

  JOIN ACADEMICO.AREAS_PRINCIPAIS APRIN
    ON APRIN.CODIGO = CREDORI.APRIN_CODIGO

  JOIN ACADEMICO.CURSOS_PG CURPG
    ON CURPG.APRIN_CODIGO = APRIN.CODIGO

  JOIN ACADEMICO.CURSOS CUR
    ON CUR.ID_CURSO = CURPG.CURSO_ID_CURSO

  JOIN CORPORATIVO.NIVEL_CURSO NIV
    ON NIV.NIVEL = CUR.NIVEL_CURSO

  JOIN ACADEMICO.PG_SIT_CRED SITCRED
    ON SITCRED.ID = CREDORI.SIT_CRED_ID

WHERE
  CREDORI.DT_TERMINO IS NULL
  AND CREDORI.SIT_CRED_ID IN (
                              8 -- CREDENCIAMENTO PONTUAL
                            , 9 -- RECREDENCIAMENTO PLENO
                            , 1 -- APOSENTADO RECREDENCIADO
                            , 3 -- CREDENCIAMENTO PLENO
                            )
ORDER BY
  PROGRAMA
```

Figura 30: SQL utilizado para obter dados dos pesquisadores. Fonte: próprio autor. São Paulo, SP. 2019

Na Plataforma Sucupira a extração dos dados foi realizada diretamente no site do Sucupira, obtido pelo endereço internet https://sucupira.capes.gov.br (pesquisa realizada em Novembro 2018).

Figura 31: Seleção do ícone Coleta - Plataforma Sucupira. Fonte: Plataforma Sucupira, Nov/2019

Através do site, seleciona-se o ícone Coleta Capes e aplica-se os filtros de para extração das informações.

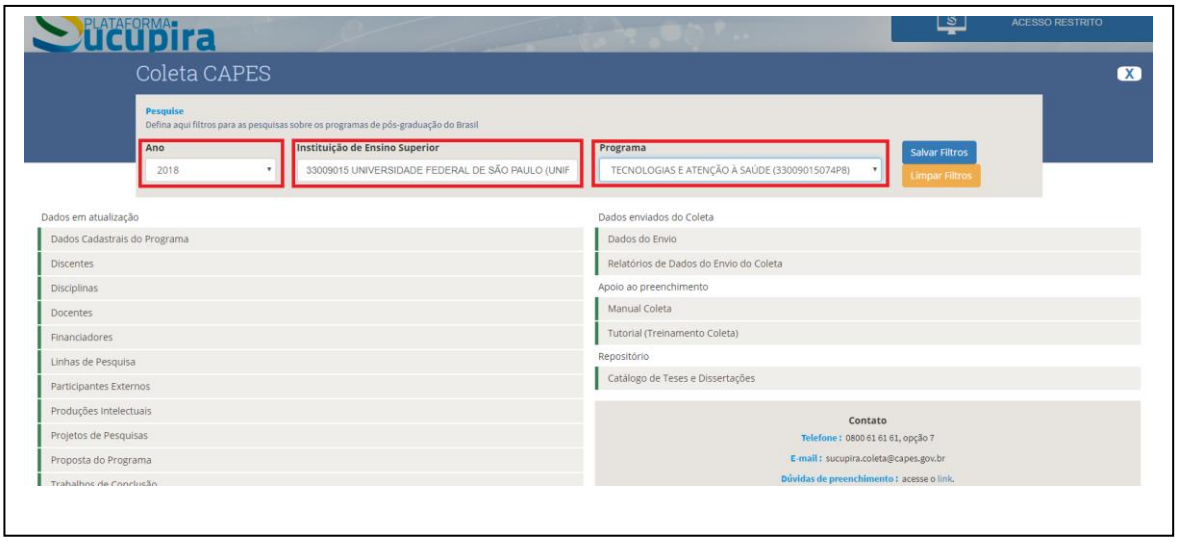


Figura 32: Criação de filtros para extração. Fonte: Plataforma Sucupira Nov/2019

Para cada Programa colocou-se o filtro ***Ano 2018***, Instituição de ensino ***33009015 Universidade Federal de São Paulo*** e Programa ***o nome do programa procurado***. A Tabela 4 contém a lista dos dez Programas de Pós-Graduação Stricto Sensu utilizados nos filtros de extração.

Os dados obtidos estão apresentados abaixo:

Pesquisadores	Total
Base de dados da UNIFESP	161
Plataforma Sucupira	160
Total sem duplicidade	149
Pesquisadores sem Lattes	2

Tabela 3: Dados obtidos das bases de dados. Fonte: próprio autor. São Paulo, SP. 2019

Observando a Tabela 3, a quantidade de pesquisadores da encontrado nas bases foi praticamente idêntica tornando assim a base de dados confiável, uma vez que a base do Sucupira é base de dados oficial validada pela CAPES nas avaliações quadrienais.

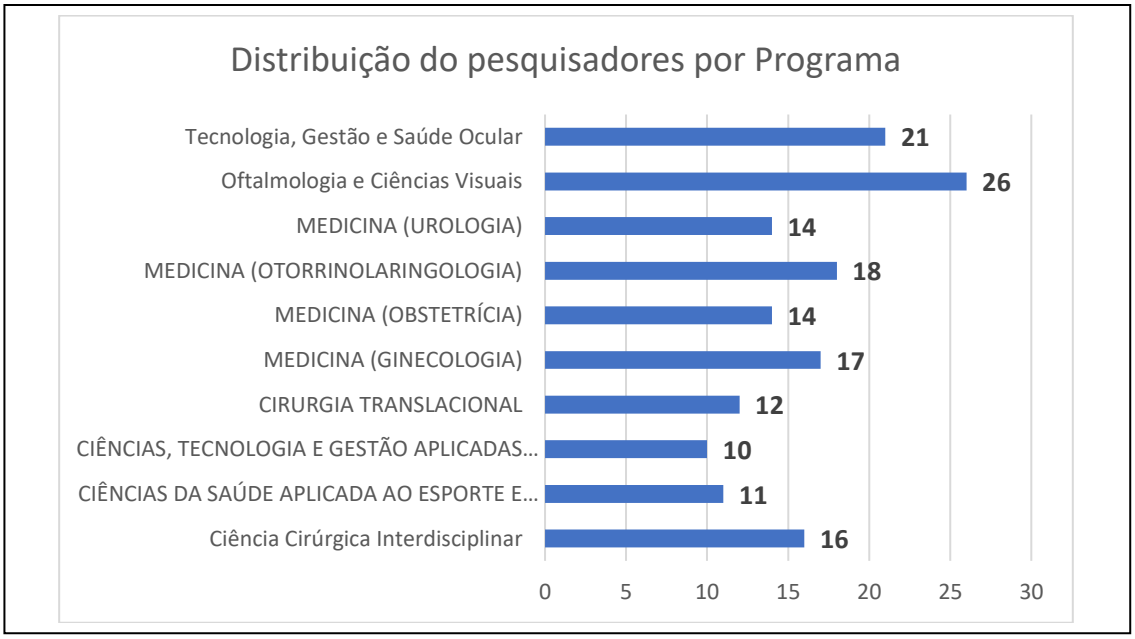


Figura 33: Distribuição de pesquisadores por Programa. Fonte: próprio autor. São Paulo, SP. 2019

A tabela acima apresenta a distribuição dos pesquisadores/orientadores dos programas da Medicina III coletados para essa pesquisa e sua distribuição nos Programas de Pós-Graduação Stricto Sensu.

6.4.6 Leitura dos dados obtidos em XML

Para leitura dos dados gerado pelo scriptLattes em XML foi desenvolvido uma aplicação escrita em linguagem Java® para o processamento dos dados recebidos.

Foi utilizado o framework Xtream (Xtream (2018) (50)) para realizar o parser no arquivo XML gerado pelo scriptLattes. Esse framework está em um repositório de código

aberto desenvolvido em Java® e é necessário fazer o download do projeto em formato “jar” e adicioná-lo ao *classpath*[12] do projeto.

Optou-se pelo Xtream pela facilidade no processo de realização de *parsing*[13] de um arquivo XML. Normalmente para se realizar o parsing de um arquivo XML utiliza-se uma *engine* ou motor de leitura e seu pós-processamento permite o acesso aos dados contidos nesse arquivo.

Figura 34: XML de saída do scriptLattes após excecução. Fonte: próprio autor. São Paulo, SP. 2019

Entretanto há várias engines que realizam esse trabalho. As mais comuns são a SAX, DOM, DOM4J, etc, cada um com características próprias e com estratégias.

6.4.7 Exportação de dados para o software Gephi

O LattesXtractor possui uma opção para salvar um arquivo no formato “.gefx”. Esse é um dos formatos de entrada para o software Gephi. Essa ferramenta permite uma representação visual de dados organizados na forma de grafos e um conjunto de algoritmos para organização e visualização dos elementos do grafo.

[12] classpath: variável de ambiente que indica onde estão localizados os arquivos binários do Java®. No Linux em no Windows ela vem definida como JAVA®_HOME e atribuída ao caminho de bibliotecas do sistema.

[13] *Parsing*: é o processo de leitura de um arquivo de computador e posteriormente realizar verificação da estrutura gramatical de forma a analisar se os dados lidos coincidem com o conjunto de regras.

6.4.8 Algoritmo utilizado na execução

Para a criação do algoritmo responsável pela organização dos dados em memória e criação dos relacionamentos entre os pesquisadores da rede de colaboração, utilizou-se de uma estrutura de classes onde fosse possível abstrair os elementos de um grafo e relacioná-los em uma rede.

Assim temos uma classe que representa o grafo completo, uma classe para representar o vértice do grafo e uma classe para representar as arestas onde são relacionados os elementos.

Após a extração o LattesXtractor relacionou os registros dos docentes para cada programa verificando se o docente já havia publicado com outro pesquisador do mesmo grupo. Se essa condição fosse satisfeita então criava-se um relacionamento entre esses pares. O código completo em Java® pode ser visualizado no Anexo 1 | Algoritmo para buscar pesquisadores e gerar XML Gephi página 98.

```
1.  DECLARA lista_pesquisadores
2.  DECLARA pesquisador_atual
3.  DECLARA pesquisador_novo
4.  PARA cada pesquisador da lista_pesquisadores FAÇA
5.      pesquisador_atual ← RECUPERA_PESQUIS (lista_pesquisadores)
6.      PARA cada pesquisador da lista_pesquisadores FAÇA
7.          pesquisador_novo ← RECUP_PESQUIS (lista_pesquisadores)
8.          SE (PUBLICA_JUNTO(pesquisador_atual, pesquisador_novo)
9.          ENTÃO
10.             INSERE_GRAFO(pesquisador_atual, pesquisador_novo)
11.         FIM SE
12.     FIM PARA
13. FIM PARA
```

Figura 35: Pseudo código para relacionamento de pesquisadores. Fonte: próprio autor. São Paulo, SP. 2019

A Figura 36 mostra como o arquivo *.gefx* é criado. A lista de elemento definida no campo “node” do XML representa os pesquisadores. Em “node” há dois atributos: id que representa o identificador do Lattes e label que identifica o nome do pesquisador (obs.: o nome está abreviado para suas iniciais por questões de privacidade).

```
<?xml version="1.0" encoding="UTF-8"?>
<gexf xmlns:viz="http:///www.gexf.net/1.1draft/viz" version="1.1" xmlns="http://www.gexf.net/1.1draft">
    <meta lastmodifieddate="2019-3-6+11:44">
        <creator>Gephi 0.7</creator>
    </meta>
    <graph defaultedgetype="undirected" idtype="string" type="static">
        <nodes count="16">
            <node id="6234829429056217" label="A.M.G."/>
            <node id="9234173201339052" label="A.G."/>
            <node id="3518607824692081" label="G.D.J.L.F."/>
            <node id="8504991156956921" label="D.M."/>
            <node id="0461653687573670" label="M.M.L."/>
            <node id="2580534578039797" label="A.A.S.N."/>
            <node id="5780852783167227" label="M.O.T."/>
            <node id="4035568020554599" label="F.A.M.H.F."/>
            <node id="0350866868370257" label="I.H.J.K."/>
            <node id="0680742555427481" label="J.L.M."/>
            <node id="8316985163665041" label="J.W."/>
            <node id="8694381071456316" label="D.J.F."/>
            <node id="0032704511396445" label="F.M.J."/>
            <node id="2871630525937037" label="H.P."/>
            <node id="8888528358909647" label="S.D.C.V.A."/>
            <node id="9796401471904195" label="R.K.S."/>
        </nodes>
        <edges count="73">
            <edge id="1" source="6234829429056217" target="9234173201339052" />
            <edge id="2" source="6234829429056217" target="3518607824692081" />
            <edge id="3" source="6234829429056217" target="6234829429056217" />
            <edge id="4" source="6234829429056217" target="8504991156956921" />
            <edge id="5" source="6234829429056217" target="0461653687573670" />
            <edge id="6" source="6234829429056217" target="2580534578039797" />
            <edge id="7" source="9234173201339052" target="5780852783167227" />
            <edge id="8" source="9234173201339052" target="9234173201339052" />
            <edge id="9" source="9234173201339052" target="3518607824692081" />
            <edge id="10" source="9234173201339052" target="6234829429056217" />
            <edge id="11" source="9234173201339052" target="8504991156956921" />
```

Figura 36: Extrutura do XML de entrada para o Gephi. Fonte: próprio autor. São Paulo, SP. 2019

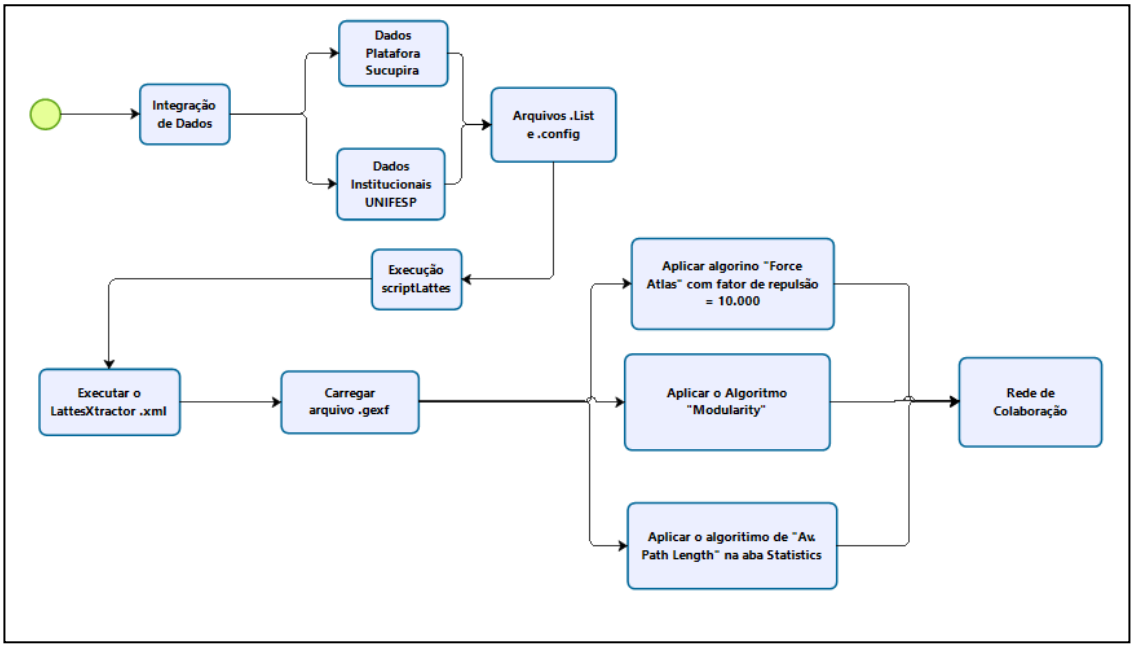


Figura 37: Processo de geração da Rede de Colaboração. Fonte: próprio autor. São Paulo, SP. 2019

A Figura 37 apresenta todo o processo de geração das redes de colaboração. O processo de importação dos dados pode ser realizado conforme descrito no

7 RESULTADOS

A seguir será apresentado os resultados das análises realizadas com a base de dados da pesquisa e os seus produtos principais. Os nomes dos pesquisadores foram apresentados somente com suas iniciais, para não haver problemas de divulgação de informação pessoal de forma imprópria e o fato de que não alterar os resultados da pesquisa.

Para cada Programa de Pós-Graduação Stricto Sensu ofertado pela Universidade Federal de São Paulo (UNIFESP), foi criado um grafo de colaboração. Um grafo de colaboração é uma imagem que representa a relação de rede de colaboração entre os pesquisadores de cada programa.

A relação entre os pesquisadores acontece toda vez que no Currículo Lattes há uma indicação de publicação em conjunto com outro pesquisador do mesmo departamento. Há então uma indicação que determina esse relacionamento.

A representação gráfica dessa relação é uma aresta, ou uma linha que liga um vértice da rede a outro vértice. Cada vértice da rede é representado por um pesquisador do respectivo departamento. Dessa forma o grafo de colaboração ou rede colaboração é apresentado na forma gráfica.

Foram extraídos currículo lattes dos professores orientadores Stricto Sensu de 10 Programas de Pós-Graduação que fazem parte da área de Medicina III e ministrados pela Universidade Federal de São Paulo.

PROGRAMA DE PÓS-GRADUAÇÃO		NÍVEIS E NOTA CAPES		
Código do Programa	Nome do Programa	ME	DO	MP
33009015009P1	CIÊNCIA CIRURGICA INTERDICIPLINAR	4	4	-
33009015173P6	CIÊNCIAS DA SAÚDE APLICADA AO ESPORTE E À ATIVIDADE FÍSICA	-	-	3
33009015093P2	CIÊNCIAS, TECNOLOGIA E GESTÃO APLICADAS À REGENERAÇÃO TECIDUAL	-	-	3
33009015038P1	CIRURGIA TRANSLACIONAL	6	6	-
33009015014P5	MEDICINA (GINECOLOGIA)	4	4	-
33009015013P9	MEDICINA (OBSTETRÍCIA)	4	4	-
33009015018P0	MEDICINA (OTORRINOLARINGOLOGIA)	4	4	-
33009015021P1	MEDICINA (UROLOGIA)	4	4	-
33009015024P0	OFTALMOLOGIA E CIÊNCIAS VISUAIS	6	6	-
33009015082P0	TECNOLOGIA, GESTÃO E SAÚDE OCULAR	-	-	3

Tabela 4: Programas pertencentes à área de Medicina III. Fonte: próprio autor. São Paulo, SP. 2019

Os dados coletados através do processo de extração das informações não permitem uma visualização de tal forma que possa ser inferido novos conhecimentos sem uma análise minuciosa.

Como forma de tentar organizar a informação foi tabulado os dados dos pesquisadores em uma tabela que apresenta seu nome e a quantidade de relações que ele possui no grupo proposto. No resultado, segue a Tabela 5 demostra os 10 primeiros elementos ordenados de forma descendente em relação ao seu grau de relacionamento (a tabela completa do grau de relacionamento pode ser vista na íntegra no Anexo 5 | Tabela de Grau de Relacionamento, pagina 124 dos anexos).

Grau de Colaboração entre Pesquisadores								
Pesquisador		Grau de relacionamento	G (0,0)	G(1,1)	G(2,5)	G(6,10)	G(11, 15)	G(16,20)
1	L.M.F.	20	0	0	0	0	0	1
2	M.D.J.S.	19	0	0	0	0	0	1
3	M.E.F.N.	17	0	0	0	0	0	1
4	M.G.F.S.	16	0	0	0	0	0	1
5	R.B.M.J.	16	0	0	0	0	0	1
6	A.L.H.D.L.F.	16	0	0	0	0	0	1
7	A.F.M.	15	0	0	0	1	0	0
8	P.S.	15	0	0	0	1	0	0
9	I.D.C.G.D.S.	14	0	0	0	1	0	0
10	E.A.J.	14	0	0	0	1	0	0

Tabela 5: Grau de colaboração aplicado em cada pesquisador. Fonte: próprio autor. São Paulo, SP. 2019

Para melhorar a visualização dos dados foi criado colunas com as relações representadas pela função G(N,M), onde

$$G(N, M) \begin{cases} N \rightarrow \text{No. mínimo de relações} \\ M \rightarrow \text{No. máximo de relações} \end{cases}$$

Assim M e N denotam a quantidade de relacionamento com outros pesquisadores

- G(0,0) → Pesquisadores que não tem relacionamento com outros pesquisadores
- G(1,1) → Pesquisadores que tem relacionamento com somente um pesquisador
- G(2,5) → Com relacionamento entre dois e cinco pesquisadores
- etc..

Com os dados tabulados obtemos a Tabela 6 com o percentual de relações encontradas o total de pesquisadores que possuem aquela relação.

	G (0,0)	G(1,1)	G(2,5)	G(6,10)	G(11, 15)	G(16,20)
%	10,74	10,07	38,26	8,72	28,19	4,03
Grau (total)	16	15	57	13	42	6

Tabela 6: Percentual de colaboração e quantitativo de pesquisadores com grau. Fonte: próprio autor. São Paulo, SP. 2019

Assim observamos que o maior percentual de pesquisadores da Medicina III da UNIFESP são aqueles que possuem entre 2 e 5 relações com outros pesquisadores totalizando 38,26% e no lado oposto, apenas 4,03% dos pesquisadores da UNIFESP é que possuem entre 10 e 20 relações. Outro dado interessante é que do universo de Programas da Medicina III da UNIFESP, 10,74% dos pesquisadores estão totalmente fora dos relacionamentos esperados. São aqueles que não publicam com nenhum outro ente do contexto dos Programas selecionados.

É muito fácil asseverar que essa sumarização dos dados ainda não consegue demonstrar de uma forma mais simplificada como ocorrem os relacionamentos entre pares de pesquisadores. Apresar de apresentar percentuais que fazem sentido e servem de base para uma avaliação analítica, ainda carece de representação mais aprofundada no que diz respeito as relações entre entes humanos.

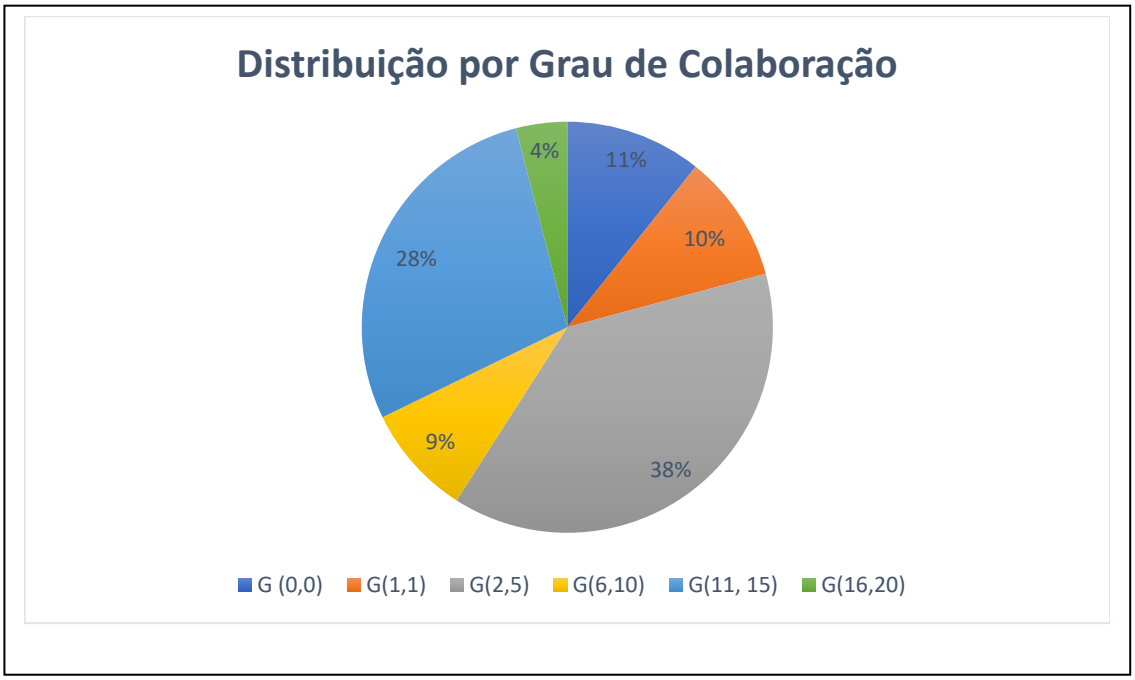


Figura 38: Distribuição dos graus de colaboração. Fonte: próprio autor. São Paulo, SP. 2019

Acima é apresentado a mesma distribuição por grau de colaboração e quantitativo de pesquisadores em um gráfico de setores. Apesar dessa representação melhorar o resultado no processo de visualização dos dados, o modelo de relacionamento proposto por relacionamento

entre pares, ainda fica deficitário e não há a riqueza de detalhes necessária para uma visualização individual de cada elemento do grupo.

Percebendo que essa deficiência de visualização e análise qualitativa é um objetivo a ser alcançado, foi então utilizado técnicas computacionais para melhorar os resultados da sumarização da Tabela 5 (Grau de colaboração aplicado a cada pesquisador)

Os algoritmos realizados com o software Gephi apresentam resultados importantes e dão ao tomador de decisão uma ferramenta de análise mais profunda. O Gephi é uma ferramenta de software capaz de gerar a rede de relacionamentos em uma estrutura de grafo. A representação de dados em um formato de grafo é ideal para observar o inter-relacionamento entre pares de objetos quaisquer ou um conjunto de objetos que mantem relação entre si.

Foi utilizado para essa técnica o arquivo de entrada gerado pelo software LattesXtractor com os dados obtidos pelo XML. No processamento dos dados do LattesXtractor realizado pela leitura do XML, foi gerado um XML intermediário com extensão de nome de arquivo ".gefx".

Com resultado dos dados contidos no arquivo .gefx e estruturados nas tags XML o Gephi é capaz de gerar imagens gráficas e sumarizar as informação em um formato de visualização e leitura fácil para estruturas correlacionadas e complexas e criar comunidades ou grupos que se inter-relacionam com maior força dentro da rede.

Para esse trabalho, na representação gráfica obtida pela Rede de Relacionamento, quanto maior o raio do vértice, maior é a indicação de que esse pesquisador se relaciona com outros pesquisadores. Isso será também percebido verificando o número de arestas (linhas) que ligam uns vértices aos outros no gráfico.

Assim, de forma simplificada um vértice na Rede de Colaboração é um circulo preenchido com uma cor e quanto maior se tamanho mais atenção ao significado deve ser analisada. Os vértices com menor tamanho também, da mesma forma devem ser investigados. Uma rede de vértices e arestas com tamanhos de nós do mesmo tamanho, pode ser considerada uma rede normalizada, ou seja, uma rede não apresenta discrepâncias em sua natureza de dados e dos elementos que o compõe.

A seguir iremos verificar os mesmos dados da Tabela 5 – Grau de colaboração aplicado a cada pesquisador, agora com a perspectiva da Rede de Colaboração e validar se a nova representação apresenta algum ganho em relação aos gráficos da Figura 38 e Tabela 6.

Ao analisarmos o resultado da rede de colaboração do Programa Medicina (Urologia) observamos também que há um pesquisador chave.

O pesquisador R.P.B. já publicou com oito outros pesquisadores de um total de catorze pesquisadores do programa. Ainda três pesquisadores não publicaram nenhum com seus pares de programa. Observamos também que o código de cores é importante nessa avaliação. Os grupos verde claros e lilás formam comunidades fazendo com que R.P.B. seja o ponto que liga esses grupos. Verificamos que M.N. e M.C. publicaram com R.P.B. e não apresentam sinergia com os outros pesquisadores do mesmo programa.

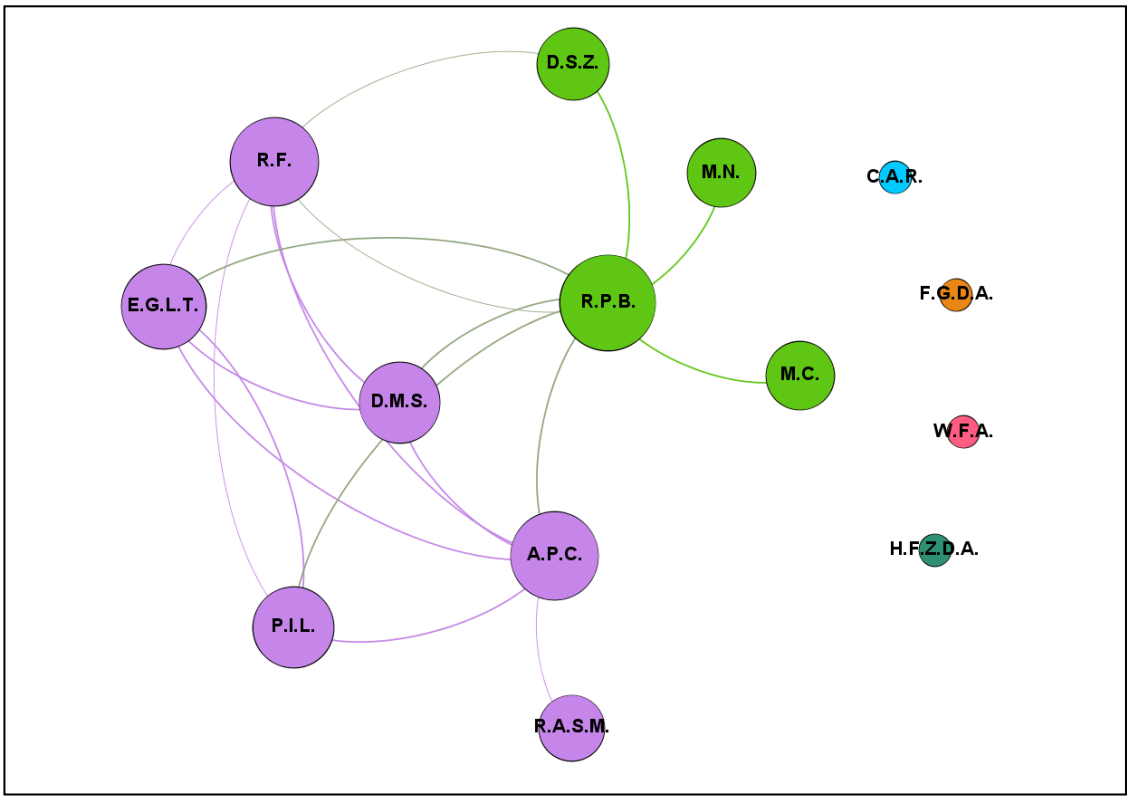


Figura 39: Rede de Colaboração de Medicina (Urologia) . Fonte: próprio autor. São Paulo, SP. 2019

Essa verificação e análise visual dá poder de velocidade na validação da informação. Rapidamente encontra-se o maior representante, seu nome e com quem ele se relaciona no grupo. Essa última premissa é muito trabalhosa de ser analisada em um quadro numérico como o apresentado na tabela que apresenta na Tabela 5: Grau de colaboração aplicado em cada pesquisador.

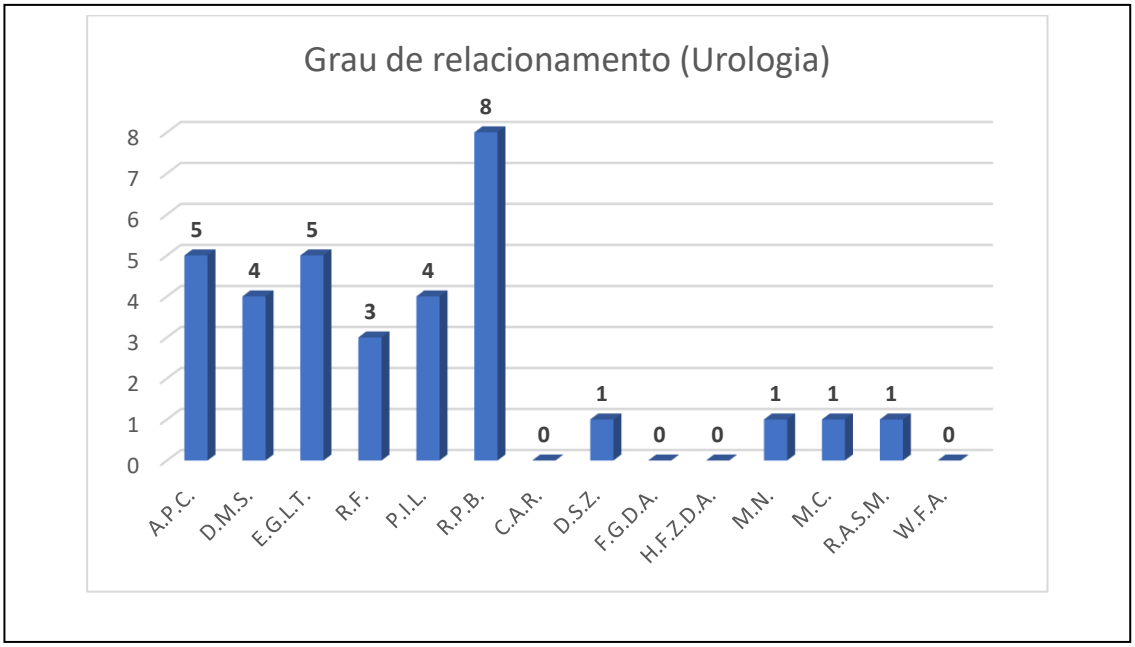


Figura 40: Grau de relacionamento Medicina (Urologia) . Fonte: próprio autor. São Paulo, SP. 2019

Veja que se tentarmos uma nova forma de visualização em um gráfico de barras como é o da Figura 40, o resultado não é rico para apresentar o conjunto de informações necessária para um relacionamento entre pares.

O gráfico acima permite demonstrar que há um elemento chave que ser relaciona muito com os demais, entretanto não apresenta de forma precisa com quais elementos ele se relaciona e o quanto ele importa nas relações. É possível deduzir que o pesquisador R.P.B. possui uma grande representatividade, mas não apresenta detalhes importantes como sua relação com os outros pesquisadores e por fim, é muito baixa a qualidade de informação sobre os demais pesquisadores da mesma rede.

A Rede de Colaboração também permite apresentar os resultados em comunidades o que nesse caso é apresentado como um conjunto de cores. Na Rede de Colaboração da Urologia observamos duas comunidades. A inscrita em cor lilás e outra inscrita em cor verde. Os elementos que não se relacionam (não publicam com outros elementos do grupo) formam comunidades independentes.

Essa relação de cores denota que tais pessoas costumam publicar entre si, então a relação é muito forte ele eles. Observa-se no gráfico que entre cada grupo existem muitas arestas em comuns. Quando um grupo se liga a outro de forma mais fraca (poucas arestas de ligação), então essa passa a ter uma cor diferente.

No exemplo da Urologia os pesquisadores R.P.B. e D.S.Z, ambos da comunidade verde, relacionam-se com outros elementos da comunidade lilás. Mas M.N. e M.C. relacionam-se somente com R.P.B. que é do grupo verde.

Essa diferenciação de cores por grupo ajuda a distinguir situações importantes em uma análise mais profunda. Nesse caso mostrou que há pesquisadores que não interagem com as pessoas do mesmo grupo e que a criação de grupos maiores seria um aspecto desejado no programa. Caberia, portanto, a um coordenador de área verificar e intervir a fim de corrigir essa situação caso ela pudesse resolver algum problema relacionado ao alinhamento dos objetivos do Programa de Pós-Graduação em questão.

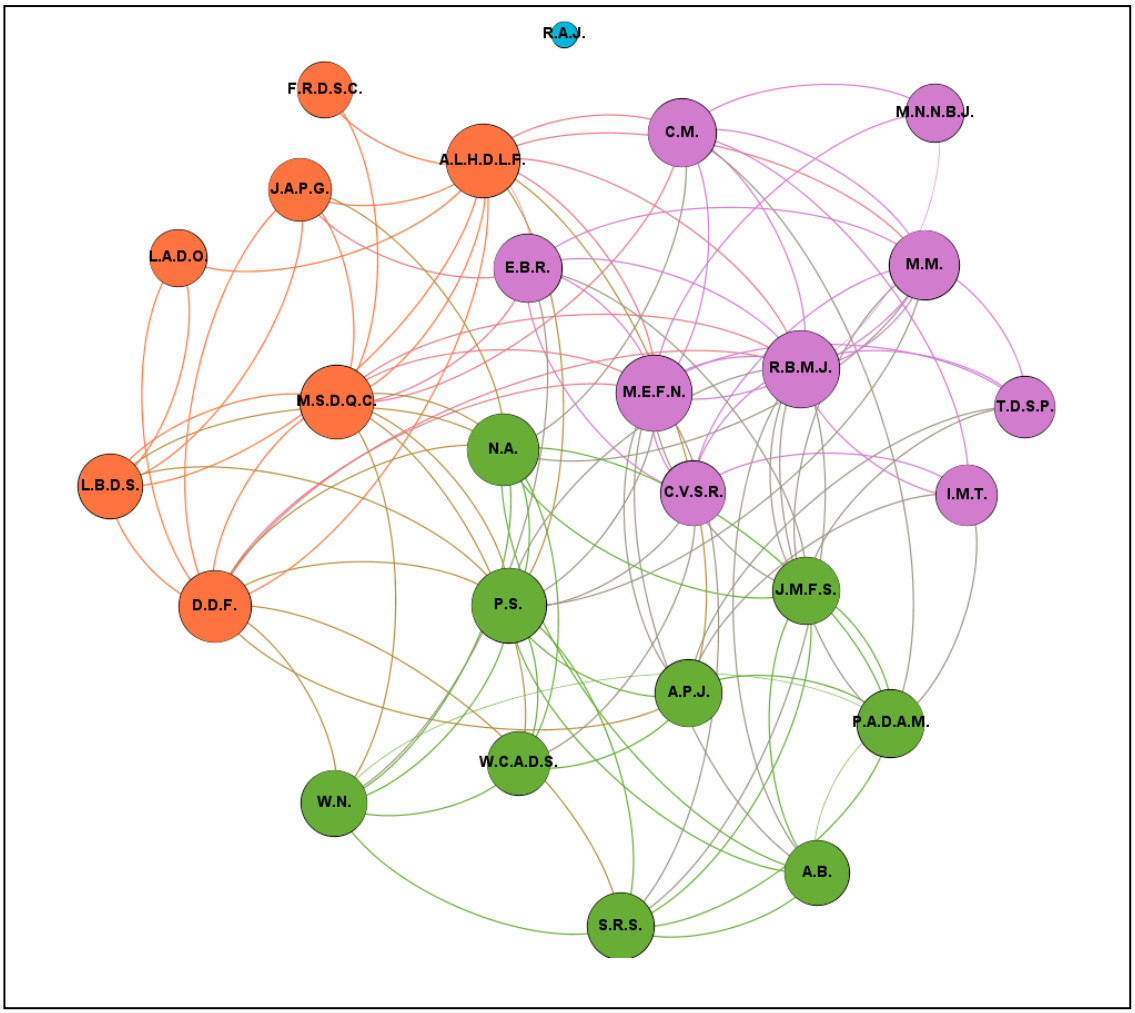


Figura 41: Rede de Colaboração Oftalmologia e Ciências Visuais. Fonte: próprio autor. São Paulo, SP. 2019

A rede de colaboração da Oftalmologia e Ciências Visuais, apresentada na Figura 41. mostra um resultado interessante de observação. É possível analisar agora um universo de

colaboração que apresenta um grau de relacionamento maior que o exemplo anterior e a análise permite um grau de hipótese maior.

Os grupos ou comunidades organizadas nas cores lilás, verde, laranja demonstram como esse grupo interage entre si. Há aqui uma discrepância importante a ser observada. Somente um pesquisador não interage com o grupo.

Isso pode ser ocasionado por alguns motivos:

- R.A.J. não está com o Lattes atualizado
- R.A.J. publica com outros pesquisadores fora do grupo ao qual pertence
- R.A.J. ainda é novo no grupo

Essas questões podem ser levantadas pelo coordenador do Programa e corrigir essa situação incomoda para que não interfira na avaliação da CAPES.

Por outro lado, observamos que o vértice que representa os pesquisadores, de uma maneira geral é muito próximo o seu tamanho. Isso é interessante pois como existe muita interação entre esse grupo é natural que não exista um pesquisador chave. Essa observação pode ser também ratificada pela quantidade de arestas que são apresentadas mostrando um grupo muito equilibrado, apesar de ainda existirem algumas comunidades.

Esse é também um típico caso de rede normalizada. Isso se traduz como um grupo que se comunica muito entre si o que torna os elementos gráficos muito próximos em relação ao seu tamanho. O alto número de arestas representada denota um grupo muito atuante. Um grupo normalizado com poucas arestas demonstrariam, ao contrário desse exemplo, um grupo de pessoas que não se relacionam entre si, e que seria objeto de medidas e tomadas de decisão para corrigir esse comportamento indesejado no que se refere a interação entre pares.

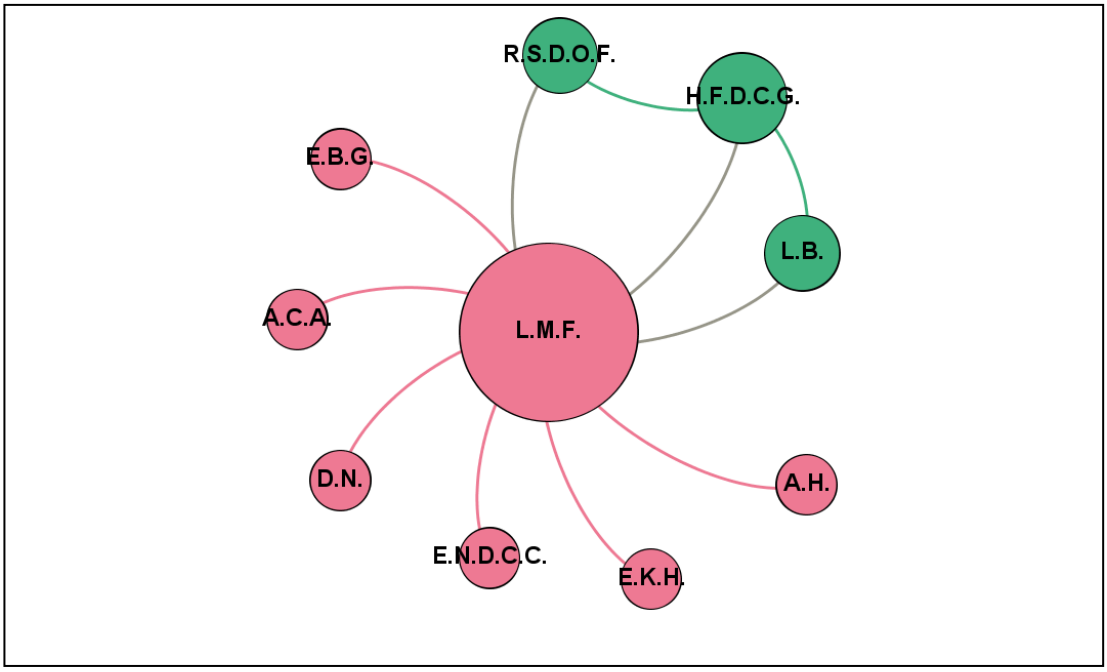


Figura 42: Ciências Tecnologia e Gestão Aplicada a Regeneração Tecidual. Fonte: próprio autor. São Paulo, SP. 2019

Na Figura 42 há também um conhecimento curioso obtido pela análise das informações processadas. Em uma análise comparativa dos elementos que compõe o gráfico em relação ao tamanho dos vértices da rede, no Programa Ciências Tecnologia e Gestão Aplicada a Regeneração Tecidual, (Figura 42) existe um pesquisador com um grau de atratividade muito grande, criando para ele forte relacionamento e para todos os outros um nível baixo de relação entre pares.

Por outro lado L.M.F. pode, ao contrário, representar uma discrepância no grupo. Todos os pesquisadores se relacionam com ele e isso pode, em um momento de avaliação, ser prejudicial ao Programa. Como um Programa de Pós-Graduação existem diversas linhas de pesquisa, é no mínimo notório que um único pesquisador possa, que já tem sua própria especialidade, publicar com outros que provavelmente não tenham a mesma especialidade.

Entretanto, talvez dentro desse Programa essa questão seja justa. Os questionamentos acima devem ser estudados dentro do universo de pesquisadores, mas a representação gráfica proposta dá poder a uma análise aprofundada baseada na interrelação visual da rede.

Como última análise, iremos observar como estão organizados os pesquisadores e seus pares de uma forma global. A Figura 43 apresenta o relacionamento de toda a comunidade de Medicina III da Universidade Federal de São Paulo – UNIFESP, com os dados coletados.

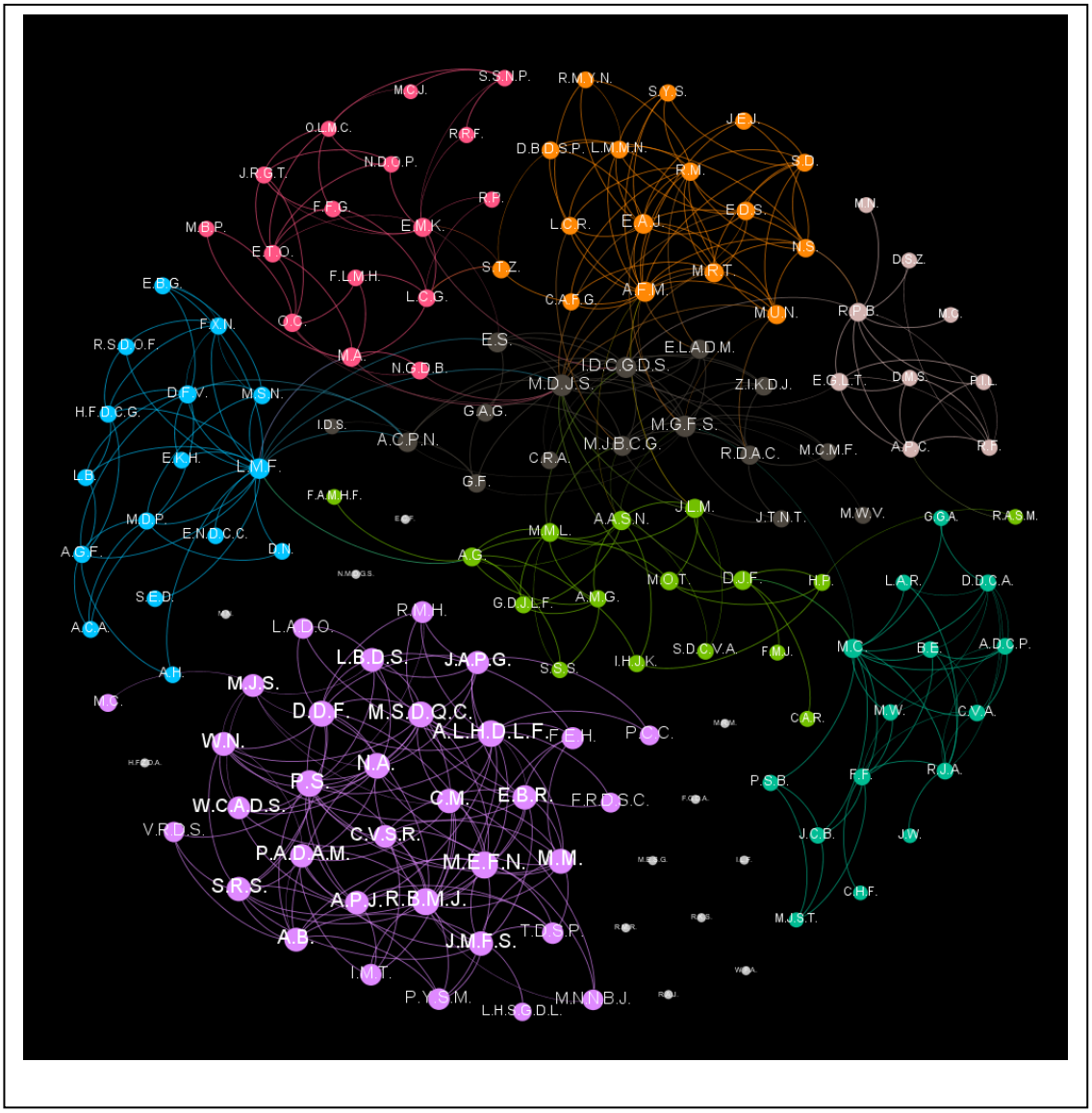


Figura 43: Rede de Colaboração da Área de Medicina III (Unifesp) . Fonte: próprio autor. São Paulo, SP. 2019

O que primeiro ponto a ser observado na representação gráfica é que existe uma sinergia entre os grupos muito grande. Poucos são os pesquisadores que não se relacional com seus pares dentro da UNIFESP, entretanto esses 11 elementos devem suscitar observação apurada por parte dos Programas de Pós-Graduação.

Atenção especial ao fato que o quadro não apresenta os grupos isolados, mas sim grupos de comunidades que interagem entre si. Uma vez que é imprescindível nos dias de hoje a interdisciplinaridade, esse comportamento é bem-vindo na universidade. Há muitas relações intercomunidades que denota a interdisciplinaridade dos pesquisadores e portanto com resultados de pesquisas com características translacionais.

Ao observarmos mais o quadro da Figura 43 em uma avaliação mais atenta, verificamos um fator interessante nos dados apresentados. Inicialmente existiam 10 programas de Pós-Graduação, mas observamos que somente oito comunidades foram representadas na forma gráfica.

Verde	Ciência Cirúrgica Interdisciplinar
Verde Musgo	Ciências da Saúde Aplicada ao Esporte e à atividade física
Cian	Ciências Tecnologia, Gestão Aplicada à Regeneração Tecidual
Cinza	Medicina (Ginecologia)
Laranja	Medicina (Obstetricia)
Vermelho	Medicina (Otorrino)
Grená	Medicina Urologia
Rosa	Oftalmologia e Ciencias Visuais

Figura 44: Medicina III organizada em comunidades de relacionamento. Fonte: próprio autor. São Paulo, SP. 2019

Os Programas Cirurgia Translacional e Tecnologia Gestão e Saúde Ocular foram atraídos para outras comunidades. Esse fator de atratividade é importante pois observa-se que eles foram aglutinados em grupos com outros programas. Isso é possível quando há um elevado grau de relacionamento dos integrantes desses programas com pesquisadores de outros programas.

Quando olhamos os grupos de forma individual não é possível essa inferência, pois trata-se de grupos isolados. Ao colocar todos os pesquisadores em uma rede, novos conhecimentos são gerados.

Também ao analisar a Rede de Colaboração da Medicina III observa-se que o grupo da Oftalmologia e Ciências Visuais (cor rosa) possui um raio de vértice que representam seus pesquisadores ligeiramente maior que o raio dos vértices da Medicina Obstetrícia (cor laranja).

Quando avaliamos os dois conjuntos separadamente, vemos que os dois possuem um grande grau de relacionamento e graficamente equivalente. Mas o fator que fez com que a Oftalmologia ficasse maior, no raio que representa seus vértices, é seu maior número de integrantes.

Os demais grupos são mais homogêneos em suas inter-relações. Poucos são os pesquisadores que não publicam com seus pares dentro dos programas a que pertencem.

Concluindo, com um potencial analítico, os dados coletados e a execução dos procedimentos propostos permitem aos Programas de Pós-Graduação tomar decisões sob aspectos importantes.

A análise dos dados permite visualizar como grupos de pesquisadores se organizam. A coordenação de grupos de pesquisa ou nos cursos de Pós-Graduação podem tirar proveito dessa análise e instigar/validar/patrocinar o resultado das correlações entre pares ao observar sua interação e assim também permita maior multiplicidade de resultados.

Identificar pessoas e seus grupos, permite direcionar melhor a condução do aproveitamento dos recursos humanos para atender a necessidades importantes da instituição.

Nos dados acima pudemos ver que existem pessoas com maior receptividade em aceitar publicar com outros pesquisadores. Esse estimulo poderia ser incentivado com aqueles que não publicam com membros de seus grupos, ampliando assim a capacidade de produção científica naquele setor.

Para os pesquisadores que se apresentam destaques de comunicação entre seus pares, a coordenação do programa pode propor discussões com eles para entender o motivo de tal desempenho e inter-relacionamento entre grupos e pessoas.

8 DISCUSSÃO

A pesquisa permitiu observar que a relação entre pares de pesquisadores é um componente indiscutível e necessário à saúde acadêmica e poder apresentar os resultados em gráficos de forma sumarizada, permite que sejam tomadas decisões importantes nos Programas e de Pós-Graduação, o que futuramente poderá nortear melhorias no processo de gestão de processos e recursos humanos com o objetivo de manter ou melhorar a nota em avaliações futuras da CAPES de tais Programas

O presente estudo atingiu seu objetivo e trouxe como resultados a exploração de informação textual para um produto de análise sumarizada e visual, que podem ser observados tanto do ponto de vista quantitativo quanto qualitativo.

Os resultados da análise podem servir para tomadas de decisão pelos Programas de Pós-Graduação para melhorar seu desempenho durante as fases de avaliação de certificação de qualidade dos Programas de Pós-Graduação promovida pela CAPES ou permitir uma visão mais detalhada de como as pesquisas cientificas estão sendo conduzidas dentro dos muros das universidades.

A utilização de representações visuais sob os paradigmas existentes permitem a amplificação do entendimento de um problema. Sob essa perspectiva, ferramentas computacionais que automatizam o trabalho se apresentam como ótimas alternativas para facilitar a compilação e organização dos dados com um nível de excelente acurácia.

Com um estudo de caso sob a temática de Redes de Colaboração entre pesquisadores e amplamente utilizada como estudo de caso nessa pesquisa, demonstrou que esta (a Rede de Colaboração) é uma técnica eficaz para análise da informação proposta permitindo a ampliação do valor da informação nas organizações e instituições, tanto em seu ambiente estratégico organizacional quanto no aspecto institucional.

Conforme Ferraz (2013) (51), o reconhecimento de que a interação de pesquisadores organizados em grupos é de vital importância pois os achados encontrados por tais equipes permitem um potencial de ampliação na performance dos resultados encontrados e geração de conhecimento que um indivíduo isolado dificilmente conseguiria.

Assim o benefício imediato que apresentamos é a investigação de grupos que impactam nos resultados dos produtos e artefatos gerados pelos catalizadores da ciência. O fenômeno da análise dos “pequenos mundos” (Balancieri (2004) (11)), apresenta de forma contundente que

os fluxos de comunicação e da interdisciplinaridade são fatores capazes de promover uma ciência com melhores resultados.

A apresentação dos dados que inicialmente são obtidos na forma de tabelas não oferecem uma forma fácil de se obter conhecimento sobre seu conteúdo. A informação relevante fica escondida no meio de textos e é difícil o entendimento do que está contido na massa de informações.

Mas com a organização proposta, podemos tirar conclusões importantes para os Programas de Pós-Graduação. Além disso, a metodologia apresenta uma forma fácil de avaliação para tomada de decisão e que pode ser replicada para qualquer outro Programa ou área da Pós-Graduação, não se restringindo à área apresentada.

Balancieri (2004) (11) apresenta que modelar e apresentar as rede de relacionamento é um desafio complexo e que seu entendimento e compreensão é um requisito importante na análise comportamental das pessoas e seus resultados em pesquisas. Essa pesquisa foi capaz de organizar os dados e apresentar resultados que mitigam os desafios expostos e apresentam de forma visual um conteúdo para rápida tomada de decisão.

São diversas as abordagens analíticas observáveis no contexto proposto. Por exemplo vimos a existência de algumas poucas particularidades que são encontradas em Programas de Pós-Graduação menores. Nesses casos existe uma tendência de que exista um personagem ou pesquisador que “carrega” um maior grau de conexão com seus pares.

É natural que nesses casos um pesquisador possua relevância entre os demais por se tratar de um Programa com poucos pares interagindo entre si. Um ótimo exemplo dessa particularidade foi o caso do programa Ciências Tecnologia e Gestão Aplicada a Regeneração Tecidual, encontrado na Figura 42 ou no Anexo 3 | Redes de colaboração por Programa. Este apresenta um grupo pequeno onde um pesquisador é o mais relevante, com todos os outros pesquisadores orbitando ao redor.

Mas com os grupos maiores essa disparidade tende sempre a estar normalizada entre os integrantes. Isso do ponto de vista das Redes de Colaboração é um fator de impacto ótimo. Demonstra sem equivoco que os docentes estão sintonizados e contribuindo uns com os outros o tempo todo.

No caso dos dados coletados e dos Programas observados percebemos essa tendência como um indicador poderoso pois demonstra que o corpo de docentes estão muito próximos no que diz respeito a pesquisa.

Importante enfatizar que as Redes de Colaboração existentes no cenário das universidades são de vital importância para a análise de Programas de Pós-Graduação, pois é uma ferramenta que permite observar resultados e o impactos resultantes recaiam na produção de conhecimento verificada pelos relacionamentos entre pares.

Com relação à ferramenta de software construída é notório perceber que esta permite a investigação dos dados propostos e, conforme Santos (2017) (38), com o crescimento da comunidade de pesquisadores fica patente que existe uma necessidade emergente em instrumentos eletrônicos precisos de análise que deem suporte a diagnósticos e investigações minuciosas à dados textuais.

A ferramenta proposta, apresenta uma síntese de explorações possíveis da informação e aponta uma forma de se pensar e avaliar um dado que se encontra de forma textual. Além disso, é também uma forma de visualizar e explorar bases de dados de forma rápida onde a visualização da informação, organizada em estruturas compreensíveis nos permitem fazer indagações sobre a perspectiva do futuro de um determinado domínio.

Conforme Santos (2017) (38), a busca, extração e tratamento de informações da Plataforma Lattes não é uma tarefa trivial. A construção de programas de computador para analisar dados brutos e obter semântica e inferência como resultado do processamento é um trabalho que é fortemente incentivado.

A exploração dos dados resultantes do software scriptLattes e pós-execução do LattesXtractor, permitiu realizar análises talvez ainda não imaginadas pelos Programas de Pós-Graduação. Outras perspectivas podem ainda ser incluídas na medida que a informação está disponível e ainda não percebidas durante o desenvolvimento dessa pesquisa.

Essa extrapolação de aumento da capacidade dos softwares envolvidos só foi possível pois houve interoperabilidade entre os sistemas. Segundo Almeida (2018) (52), é uma necessidade do mundo atual transformar objetos brutos em elementos de informação. Assim a integração de diversos sistemas deve ser uma meta a ser cada vez mais buscada. Cada software desenvolvido atende a uma especificidade e unir os resultados de processamento para obtenção de um melhor resultado trará mais benefícios aos usuários.

Finalizando, é notório que a construção da instrumental tecnológico que dê suporte a processamento de informações que resulte em análise de padrões de comportamento do ponto de vista qualitativo e quantitativo. É uma necessidade para as organizações e instituições tais instrumentos, pois os mesmos podem apresentar resultados confiáveis e assim auxiliar na avaliação de que o recurso investido pelas agências de fomento está atendendo as expectativas da áreas de pesquisa que emprega tais recursos.

9 CONCLUSÃO

Com o método realizado nessa pesquisa foi possível construir uma ferramenta com capacidades de explorar informações textuais e produzir conhecimento útil para tomadas de decisão. A eficácia da ferramenta foi demonstrada através do resultados apresentados, trazendo à pesquisa que tem como foco na análise informações baseado no Currículo Lattes uma nova técnica para abordagem para o tipo de dado apresentado.

A integração da ferramenta com o scriptLattes também é uma contribuição importante. Como o scriptLattes é uma ferramenta de grande utilização no meio acadêmico para obter informações do Lattes, é de grande valor construir instrumento tecnológico que se conecte e realize o a interoperabilidade entre as ferramentas já existentes, estendendo a capacidade das potencialidades das já existentes.

Ainda a integração de ferramentas com o software especialista de análise de redes Gephi foi capaz de examinar com acuidade a relações de pesquisadores utilizando funcionalidades e algoritmos próprios de exploração de redes, filtragem, navegação e agrupamento de dados o que é bom para encontrar padrões facilitando análises complexas.

O estudo de caso apresentado, pode, com sucesso, exemplificar o potencial da ferramenta desenvolvida e demonstrou o potencial para que outros estudos sejam criados, estendendo as potencialidades da atual ferramenta para se extrair conhecimento em análises que dependam da informação estruturada do Lattes.

O LattesXtractor demostrou esse processo através de uma interface de programação simples que obtém dados baseados em XML e o transforma em conteúdo em memória pronto para ser consumido por ferramentas de estatística, visualizações de dados, geração de relatórios, exportação para outros formatos, etc. Nesse trabalho a exportação e geração do arquivo de entrada de forma automatizada para o software Gephi foi realizado com sucesso.

Concluindo, não seria lógico limitar ou esgotar as possibilidades da ferramenta. Com o advento do Big Data, está cada vez mais claro que será necessário criar novas perspectivas com os dados existentes. Os resultados apresentados até aqui demonstraram que esse campo de análise ainda permite novas explorações, pois existem muitas lacunas que podem ser analisadas e preenchidas.

10 LIMITAÇÕES DA PESQUISA

Esse capítulo apresenta algumas limitações da pesquisa e os aspectos que não foram explorados. Importante lembrar que a Plataforma Lattes é uma plataforma aberta e seu preenchimento é de responsabilidade do próprio pesquisador.

Apesar do importante desafio alcançado pelo CNPq em manter um banco de currículos de cientistas, a forma do preenchimento dos Currículo Lattes pode impactar os resultados.

Assim, nem todos os pesquisadores brasileiros ou estrangeiros com pesquisa no Brasil tem o Currículo Lattes. Sobre o preenchimento das informações, muitas vezes estas podem não estar corretas ou estar incompletas. Isso gera um prejuízo na avaliação da informação encontrada no repositório, porém não é um fator que permita o descarte de análises.

As análises, no atual estado da arte da aplicação, estiveram restritas aos resultados do processamento do scriptLattes. Apesar do scriptLattes realizar um esforço enorme na obtenção dos dados, observamos que em alguns casos seria necessário ainda mais um nível de refinamento e de pós-tratamento. Escolhemos não realizar análises sofisticadas pois os dados obtidos já permitiam alcançar, com nível aceitável e alta taxa de acerto, os objetivos propostos. Entretanto é possível tratar a informação aplicando algoritmos mais robustos e assim gerar mais conhecimento e resultados importantes.

11 TRABALHOS FUTUROS

O LattesXtractor pode é apenas uma “ponta no iceberg” de ferramentas que extraem e compilam informações. Até o atual ponto de desenvolvimento a ferramenta permitiu exportar dados de uma ferramenta para outra e operar de forma compartilhada com outros softwares.

Existem ainda muitas lacunas que podem ser exploradas. No atual estado da arte, a ferramenta se alimenta de informações de softwares terceiros. Uma perspectiva é criar um motor próprio de extração de dados da Plataforma Lattes e integrar novos componentes, como uma base de dados relacional e alimenta-la com os dados extraídos permitindo criar assim bases históricas para acompanhamento da evolução da informação cientifica.

Também é possível a criação e exportação de dashboards completos com os perfis dos pesquisadores, suas pesquisas e alunos em um formato pronto para a Web podendo ser consumidos diretamente por portais institucionais com informações relevantes e que atendam a nichos específicos dos Programas.

O *scriptLattes* já cumpre parte dessa tarefa. Entretanto a proposta é incluir novas formatações de apresentação e resultados que atendam a expectativas próprias de cada instituição adequando-se a layouts existentes com técnicas de padronização aplicadas a CSS (*Cascading Style Sheets*)[14].

Outra lacuna são as nuvens de *tags* para encontrar conhecimento baseado em publicações realizadas. As publicações em revista dos pesquisadores são um ótimo artefato de análise, uma vez que nesse produto estão descritos, além das pesquisas em si, também o tipo de trabalho especialista que o pesquisador realiza. Assim a exploração do texto pode trazer a relevância de um conjunto de trabalhos e apontar onde estão os polos de conhecimento de determinado grupo de pesquisa.

[14] Técnica para aplicar estilos (cores, fontes, espaçamento, disposição de elementos na tela) a um documento web.

REFERÊNCIAS BIBLIOGRÁFICAS

1. CAPES. CAPES - Coordenação de Aperfeiçoamento de Pessoal de Nível Superior. [Online].; 2018. Available from: http://www.capes.gov.br.

2. Bonifácio A. S.. Ontologia e consulta semântica: uma aplicação ao Caso Lattes. Tese de dissertação de mestrado pela UFRGS. 2002.

3. Lima J. C. eA. Ontologias - OWL (Web Ontology Language). Universidade Federal de Goias (UFG). 2005.

4. Berners-Lee HJTLO. The Semantic Web. Issue of Scientific American. 2001.

5. Breitman K. K.. Web Semâtica – A Internet do Futuro. LTC Editora. 2005.

6. Ribeiro R. J.. CAPES. [Online].; 2013. Available from: http://www.fmrp.usp.br/cpg/arquivos/Artigo_18_07_07%5B1%5D.pdf.

7. Machado R. N.. A análise cientométrica dos estudos bibliométricos publicados em periódicos da área de biblioteconomia e ciência da informação. 2005.

8. Francisco RP. Gestão de Redes de Colaboração: Conceitos e Aplicações. Facunicamps. 2018.

9. Balancieri R, Bovo AB, Kern VM, Pacheco RCdS, Barcia RM. A análise de Redes de Colaboração Científica sob as novas tecnologias de informação e comunicação: um estudo na Plataforma Lattes. CI. Inf. Brasilia. 2005;: p. 64-77.

10. Maia MdFS, Zanotto SR, Caregnato SE. Colaboração cientifica e análise de redes sociais. Revista do Instituto de Ciências Humanas e da Informação. 2011;: p. 43-55.

11. Balancieri R. Análise de redes de pesquisa em uma plataforma de gestão em ciência e tecnologia: uma aplicação à Plataforma Lattes. Dissertação de Mestrado em Engenharia de Produção - Universidade Federal de Santa Catarina. 2004.

12. Cervantes EP. Análise de Redes de Colaboração Ciêntifica: Uma abordagem baseada em grafos relacionais com atributos. Dissertação de Mestrado. São Paulo: Unversidade de São Paulo, Instituto de Matemática e Estatística; 2015.

13. Recuero R. Análise de Redes Sociais online Salvador: Edufba; 2017.

14. Smith M. The trend toward multiple authorship in pycology. American Psychologist. 1958;: p. 596-599.

15. Oliveira WAd. Colaboração científica nos programas de Pós-Graduação em Educação: uma análise de redes de coautoria. São Carlos: Universidade Federal de São Carlos, Programa de Pós-Graduação em Educação UFSC; 2017.

16. Magalhães FLFd, Garcia RDR, Souza CCGd, Sartoratto RdS, Franco ECCP, Gaspar MA. Panorama Quantitativo dos Programas de Pós-graduação Stricto Sensu em Tecnologia da Informação no Brasil. Revista Iberoamericana de Tecnología en Educación y Educación en Tecnología. 2018 Mar.

17. Raynault C. OS DESAFIOS CONTEMPORÂNEOS DA PRODUÇÃO DO CONHECIMENTO: O APELO PARA INTERDISCIPLINARIDADE. INTERthesis. 2014; 11(1).

18. Oliveira FBd. Origem e evolução dos cursos de Pós-Graduação Lato Sensu no Brasil. RAP. 1995 Jan/Mar.

19. Silva TC, Bardagi MP. O aluno de pós-graduação stricto sensu no Brasil: revisão da literatura dos últimos 20 anos. RBPG. 2015 Dec; 12.

20. PIMENTEL RG. E agora José?: jovens psicólogos recém-formados no processo de inserção no mercado de trabalho. Dissertação de mestrado. Florianópolis: Universidade Federal de Santa Catarina, Programa de Pós-Graduação em Psicologia; 2017.

21. Lievore C, Picinin CT, Pilatti LA. As áreas do conhecimento na pós-graduação stricto sensu brasileira: crescimento longitudinal entre 1995 e 2014. Ensaio: aval. pol. públ. Educ. 2017 jan/mar; 25.

22. Guimarães JA, Almeida EC. Brazil's growing production of scientific articles: how are we doing with review articles and other qualitative indicators? Scientometrics. 2013 Nov; 97(2).

23. Resende DA. Engenharia de Software e sistemas de informação. 3rd ed. Brasport , editor. Rio de Janeiro: Brasport Livros e Multimidia Ltda; 2005.

24. Pressman RS, Maxim BR. Engenharia de Software: uma abordagem profissional. 8th ed. Bookman , editor. Porto Alegre: Mc Graw Hill; 2016.

25. Royce WW. Managing the Development of Large Software Systems. IEEE. 1970 Aug: p. 1-9.

26. Royce WW. Wikipedia. [Online].; 2018 [cited 2018 12 28. Available from: https://en.wikipedia.org/wiki/Winston_W._Royce.

27. Valente C. Engenharia de Software. 2008. Módulo da Disciplina de Engenharia de Sistemas - ESAB - Escola Superior Aberta do Brasil.

28. Santos IR. Modelo para processo de desenvolvimento de software a partir da Engenharia de Requisitos: Uma propotipação orientada a empresas do APL de TI do Sudoeste do Paraná. Dissertação de Mestrado. Pato Branco: Universidade Tecnológica Federal do Paraná, Programa de Pós-Graduação em Engenharia de Produção e Sistemas; 2016.

29. Lessa RO, Lessa Junior EO. Modelos de Processos de Engenharia de Software. 2015.

30. Roy PV. Programming Paradigms for Dummies - What every programmer should know. In Programing Paradigm.; 2009.

31. Biondo G. Um processo de conversão de sistemas legado procedurais para orientado a objetos, direcionado pela arquitetura MVC. Dissertação de Mestrado. Bento Gonçalves: Universidade de Duque de Caxias; 2017.

32. Deitel HM, Deitel PJ. Java como Programar. 10th ed. Perason , editor. São Paulo: Perason; 2016.

33. Puga S. Lógica de Programação e Estrutura de dados com aplicações Java São Paulo: Pearson Education do Brasil; 2004.

34. W3C. W3C. [Online].; 2018 [cited 2018 12 28. Available from: https://www.w3.org/TR/xml/.

35. Ray ET. Learning XML. 8th ed. O'Reilly , editor. Sebastopol: O'Reilly & Associates, Inc; 2003.

36. Fugeri S. Business to Business: Aprenda a desenvolver aplicações São Paulo: Érica; 2001.

37. Siqueira MA. XML na Ciência da Informação: uma análise do MARC 21. Dissertação de Mestrado. Marilia: Universidade Estadual Paulista - UNESP, Programa de Pós-Graduação em Ciência da Informação; 2003.

38. Santos SDd. Método de Agrupamento Hieráquico a partir de Curriculos Acadêmicos, Dissertação (Mestrado). ; 2017.

39. Groner L. Groner. [Online].; 2009 [cited 2019 01 09. Available from: https://loiane.com/2009/04/construindo-um-dtd-e2-80-93-introducao-ao-xml-parte-vi/.

40. Eduardo E. Dev Media. [Online].; 2012 [cited 2019 01 27. Available from: https://www.devmedia.com.br/processamento-de-xml-em-java-com-a-api-sax/25061.

41. Claro DB, Sobral JBN. Programação em Java Departamento de Informática e Estatística CUFdSC, editor. Santa Catarina; 2000.

42. Cavalcanti TG, Almeida VCd. Caracterização do uso de construções da liguagem Java em Projetos Open Source. Tese de Conclusão de Curso. Brasilia - Distrito Federal:, Departamento de Ciência da Computação; 2016.

43. Oracle. Oracla - The Java Source. [Online].; 2019 [cited 2019 01 20. Available from: https://blogs.oracle.com/java/jdk-8-early-access-developer-documentation-updates.

44. Venners B. Inside the Java Virtual Machine. 2nd ed. Sunnyvale, California: Computing McGraw-Hill; 2000.

45. Lindholm T, Yellin F, Bracha G, Buckley A. The Java® Virtual. Specification: JSR-337 Java® SE 8 Release Contents. California: Oracle America, Inc., Oracle America, Inc. and/or its affiliates; 2015.

46. Rangel F, Silva AFd. A MÁQUINA VIRTUAL JAVA E A OTIMIZAÇÃO INLINE: UM ESTUDO DE CASO. Revista Tecnológica de Maringa. 2012: p. 103-118.

47. Gupta S, Bhatia K. A Comparative Study of Hidden Web Crawlers. International Journal of Computer Trends and Technology. 2014 Jun; 12(3).

48. Reis T. Algoritmo rastreador web especialista nuclear. Dissertação Mestrado. São Paulo: Instituto de Pesquisas Enegéticas e Nucleares - IPEN - USP, Tecnologia Nuclear e Reatores; 2013.

49. Alves AD, Yanasse HH, Soma NY. Lattes Miner: uma linguagem de domínio específico para extração automática de informações da Plataforma Lattes. In XIII Workshop de Computação Aplicada - WORCAP 2012; 2012; São Jose dos Campos.

50. Xstream. Xstream. [Online].; 2018. Available from: http://x-stream.github.io/.

51. Ferraz RRN. A utilização da ferramenta computacional Scriptlattes para avaliação das competências em pesquisa no Brasil. Prisma. 2013;: p. 41.

52. Almeida CS. Integração de dados geográficos na interoperabilidade de sistemas de apoio a decisão. Dissertação de mestrado. Coimbra - Portugal: Universidade de Coimbra, Faculdade de Ciências e Tecnologia; 2018.

53. Mena-Chalco JP. scriptLattes: Anopen-source knowledge extraction system from the Lattes platform. Journal of the Brazilian Computer Society. 2009;: p. 31-19.

54. Mena-Chalco JP. Prospecção de dados acadêmicos de currículos Lattes através de scriptLattes. In Bibliometria e Cientometria: reflexões teóricas e interfaces.: Pedro & João Editores; 2013. p. 109-128.

55. Souza R. R. AL. A Web Semântica e suas contribuições para a ciência da informação. Ci. Inf., Brasília. 2004;: p. 132-141.

56. Ferraz RRN, QUONIAM L. A Perspectivas em Gestão & Conhecimento. João Pessoa. 2014;: p. 133-147.

57. Newnan ME. Coauthorship networks and patterns of scientific collaboration. Proceedings of the National Academy of Sciences of the United States of America. 2004;: p. 5200-5205.

58. Plataforma Lattes. [Online].; 2018. Available from: http://lattes.cnpq.br/ acessado em 13/02/2018.

59. Souza CGBBGLLS. Redes de Colaboração Cientifica na área de Quimica no Brasil: Um estudo baseado nas coautoriais dos artigos da revista Quimica Nova. Departamento de Engenharia de Produção, Centro Federal de Educação Tecnológica Celso Suckow da Fonseca. 2017.

60. Giordano DM. Uso do scriptLattes e Gephi na Análise da Colaboração. Computer on the Beach. 2015;: p. 239-248.

61. Francisco RdP. Gestão de redes de Colaboração: Conceitos e Aplicações. FACUNICAMPS, Núcleo de Pesquisa e Extensão. 2011.

62. Andrade FS. Java Virtual Machine em FPGA. Trabalho de Conclusão de Curso. Curitiba: Universidade Positivo, Departamento de Engenharia da Computação; 2008.

63. Digiampietri LA, Mena-Chalco JP. Minerando e caracterizando dados de Curriculo Lattes. 2011. Centro de Matemática, Computação e Cognição da Universidade Federal do ABC.

64. Salomé BR, Nunes FLS, Chaim ML. Gerador de Sítios de Grupos de Pesquisa com inclusão automática de conteúdo baseada na Plataforma Lattes. In

VIII Simpósio Brasileiro de Sistemas de Informação (SBSI 2012); 2012; São Paulo. p. 12.

65. Gama IdS, Carvalho LdS. Tendências e perspectivas de pesquisa sobre repositórios digitais no Brasil: uma análise de Rede Sociais (ARS). Reciis – Rev Eletron Comun Inf Inov Saúde. 2017 Nov; e-ISSN 1981-6278.

66. Granjeiro RdR, Pinto AMG, Vinhas FD. Rede de colaboração cientifica: uma análise das publicações do encontro nacional de pesquisadores em gestão social. Cardenos Gestão Social. 2013 Dec; 4(2).

67. Khan M, Khan SS. Data and Information Visualization Methods, and Interactive Mechanisms: A Survey. International Journal of Computer Applications. 2011 Nov; 34.

68. Fry B. Visualizing Data. 1st ed. Oram A, editor. New York: O'Reilly Media, Inc; 2008.

69. Grus J. Data Science from Scratch. First Edition ed. Beaugureau M, editor. New York: O'Reilly Media, Inc.; 2015.

70. Ware C. Information Visualization Buehler M, editor.: Elsevier Inc. - Morgan Kaufmann publications; 2004.

71. Magalhães FLFd, Silva LCd, Gaspar MA, Carvalho AC, Mauro MH. Gestão do conhecimento: Estudo da produção de teses e dissertações Brasileiras (2006-2015). Didáctica y Educación. 2018 Nov; IX.

72. Silva LCd, Campanelli A, Silva LSd, Silva TVFd, Silva RC, Garcia RDR, et al. Graduação em TI no Brasil: perspectiva a partir do Exame Nacional de Avaliação de Desempenho. Revista Iberoamericana de Tecnología en Educación y Educación en Tecnología. 2018.

73. Wikipedia DAK. Wikipedia. [Online].; 2018 [cited 2018 11 02. Available from: https://pt.wikipedia.org/wiki/Sete_pontes_de_K%C3%B6nigsberg.

APÊNDICE

Anexo 1 | Algoritmo para buscar pesquisadores e gerar XML Gephi

```
1.  public void createXml(CurriculoLattes cv, String path, boolean nomePesquisadorCompleto) {
2.      /* Definição da estrutura */
3.      Graph grapho    = null;
4.      Node nodeSource = null;
5.      Node nodeTarget = null;
6.      Edge edge       = null;
7.      List <String> colaboradores = null;
8.      grapho = new Graph();
9.
10.     /*
11.      * Recupera os nos e arestas
12.      */
13.     para cada Pesquisador no objeto CurriculoLattes faç c  (int i = 0; i < cv.getPesquisador().size(); i++){
14.             Pesquisador p = cv.getPesquisador().get(i);
15.             //System.out.println("Nome: " + p.getIdentificacao().getNomeCompleto());
16.             nodeSource = new Node();
17.
18.             // Codigo Lattes
19.             nodeSource.setId(p.getId());
20.
21.             String nomePesquisador = p.getIdentificacao().getNomeCompleto();
22.
23.             if (!nomePesquisadorCompleto) {
24.                     nomePesquisador = XtractorHelper.firstLetterName(nomePesquisador);
25.             }
26.
27.             nodeSource.setLabel(nomePesquisador);
28.             grapho.addNode(nodeSource);
29.
30.             // Recupera as relacoes
31.             if (p.getColaboradores() != null) {
32.                     colaboradores = p.getColaboradores().getColaboradores();
33.                     for (int k = 0; k < colaboradores.size(); k++){
34.
35.                             nodeTarget = new Node();
36.                             nodeTarget.setLabel("");
37.                             nodeTarget.setId(colaboradores.get(k));
38.
39.                             //#####################################################################
40.                             //Verifica se o pesquisador esta na lista de curriculuns recuperados
41.                             for (int y = 0 ; y < cv.getPesquisador().size(); y++) {
42.                                     if (cv.getPesquisador().get(y).getId().equals(nodeTarget.getId())){
43.
44.                                             nomePesquisador                                         =
cv.getPesquisador().get(y).getIdentificacao().getNomeCompleto();
45.                                             if (!nomePesquisadorCompleto) {
46.                                                     nomePesquisador                                 =
XtractorHelper.firstLetterName(nomePesquisador);
47.                                             }
48.
49.                                             nodeTarget.setLabel(nomePesquisador);
50.                                             //System.out.println("                    " +nodeTarget.getId() + " : " +
nodeTarget.getLabel() );
51.                                             edge = new Edge(nodeSource, nodeTarget);
52.                                             grapho.addEdge(edge);
53.                                             grapho.addNode(nodeTarget);
54.                                             break;
55.                                     }
56.                             }
57.
58.
59.                     }
60.             } else {
61.                     grapho.addNode(nodeSource);
62.             }
63.     }
64.
65.     grapho.generateGephiGraphXml(path);
66. }
```

Anexo 2 | Lista completa de currículos analisados

MEDICINA III - EXTRAÇÃO DE DADOS EM 11/2018			
DADOS DO PROGRAMA DE PÓS-GRADUAÇÃO		DOCENTE - ORIENTADOR CREDENCIADO	
CODIGO CAPES	PROGRAMA	NOME	CODIGO LATTES 16 DIGITOS
33009015009P1	Ciência Cirúrgica Interdisciplinar	ADRIANO MIZIARA GONZALEZ	6234829429056217
33009015009P2	Ciência Cirúrgica Interdisciplinar	ALBERTO GOLDENBERG	9234173201339052
33009015009P3	Ciência Cirúrgica Interdisciplinar	ALCIDES AUGUSTO SALZEDAS NETTO	2580534578039797
33009015009P4	Ciência Cirúrgica Interdisciplinar	DJALMA JOSE FAGUNDES	8694381071456316
33009015009P5	Ciência Cirúrgica Interdisciplinar	FAUSTO MIRANDA JUNIOR	0032704511396445
33009015009P6	Ciência Cirúrgica Interdisciplinar	FERNANDO AUGUSTO MARDIROS HERBELLA FERNANDES	4035568020554599
33009015009P7	Ciência Cirúrgica Interdisciplinar	GASPAR DE JESUS LOPES FILHO	3518607824692081
33009015009P8	Ciência Cirúrgica Interdisciplinar	HELIO PLAPLER	2871630525937037
33009015009P9	Ciência Cirúrgica Interdisciplinar	IVAN HONG JUN KOH	0350866868370257
33009015009P10	Ciência Cirúrgica Interdisciplinar	JAQUES WAISBERG	8316985163665041
33009015009P11	Ciência Cirúrgica Interdisciplinar	JOSE LUIZ MARTINS	0680742555427481
33009015009P12	Ciência Cirúrgica Interdisciplinar	MARCELO MOURA LINHARES	0461653687573670
33009015009P13	Ciência Cirúrgica Interdisciplinar	MURCHED OMAR TAHA	5780852783167227
33009015009P14	Ciência Cirúrgica Interdisciplinar	RIOKO KIMIKO SAKATA	9796401471904195
33009015009P15	Ciência Cirúrgica Interdisciplinar	SARHAN SYDNEY SAAD	8646840760424911
33009015009P16	Ciência Cirúrgica Interdisciplinar	SIMONE DE CAMPOS VIEIRA ABIB	8888528358909647
33009015173P6	CIÊNCIAS DA SAÚDE APLICADA AO ESPORTE E À ATIVIDADE FÍSICA	ALBERTO DE CASTRO POCHINI	2476659894036430
33009015173P7	CIÊNCIAS DA SAÚDE APLICADA AO ESPORTE E À ATIVIDADE FÍSICA	BENNO EJNISMAN	1124807952912223
33009015173P8	CIÊNCIAS DA SAÚDE APLICADA AO ESPORTE E À ATIVIDADE FÍSICA	CARLOS HENRIQUE FERNANDES	0304502442424862
33009015173P9	CIÊNCIAS DA SAÚDE APLICADA AO ESPORTE E À ATIVIDADE FÍSICA	CARLOS VICENTE ANDREOLI	0079323365496289
33009015173P10	CIÊNCIAS DA SAÚDE APLICADA AO ESPORTE E À ATIVIDADE FÍSICA	DIEGO DA COSTA ASTUR	6230616550486112
33009015173P11	CIÊNCIAS DA SAÚDE APLICADA AO ESPORTE E À ATIVIDADE FÍSICA	GUSTAVO GONCALVES ARLIANI	6509085647085311
33009015173P12	CIÊNCIAS DA SAÚDE APLICADA AO ESPORTE E À ATIVIDADE FÍSICA	IRINEU LOTURCO FILHO	0370724009248558
33009015173P13	CIÊNCIAS DA SAÚDE APLICADA AO ESPORTE E À ATIVIDADE FÍSICA	LEONARDO ADDEO RAMOS	7752197615598277
33009015173P14	CIÊNCIAS DA SAÚDE APLICADA AO ESPORTE E À ATIVIDADE FÍSICA	MARCELO WAJCHENBERG	4615316496474251
33009015173P16	CIÊNCIAS DA SAÚDE APLICADA AO ESPORTE E À ATIVIDADE FÍSICA	MOISES COHEN	6174355233304675

33009015173P17	CIÊNCIAS DA SAÚDE APLICADA AO ESPORTE E À ATIVIDADE FÍSICA	PAULO SANTORO BELANGERO	0399504221133550
33009015093P2	CIÊNCIAS, TECNOLOGIA E GESTÃO APLICADAS À REGENERAÇÃO TECIDUAL	ALESSANDRA HADDAD	0881683207035609
33009015093P3	CIÊNCIAS, TECNOLOGIA E GESTÃO APLICADAS À REGENERAÇÃO TECIDUAL	ANTONIO CARLOS ALOISE	9137246134947408
33009015093P4	CIÊNCIAS, TECNOLOGIA E GESTÃO APLICADAS À REGENERAÇÃO TECIDUAL	DENISE NICODEMO	5221369826598914
33009015093P5	CIÊNCIAS, TECNOLOGIA E GESTÃO APLICADAS À REGENERAÇÃO TECIDUAL	ELAINE KAWANO HORIBE	5423291926154188
33009015093P6	CIÊNCIAS, TECNOLOGIA E GESTÃO APLICADAS À REGENERAÇÃO TECIDUAL	ELIZIANE NITZ DE CARVALHO CALVI	4190553604440999
33009015093P7	CIÊNCIAS, TECNOLOGIA E GESTÃO APLICADAS À REGENERAÇÃO TECIDUAL	ELVIO BUENO GARCIA	5558512244477786
33009015093P8	CIÊNCIAS, TECNOLOGIA E GESTÃO APLICADAS À REGENERAÇÃO TECIDUAL	HEITOR FRANCISCO DE CARVALHO GOMES	2266460253828291
33009015093P9	CIÊNCIAS, TECNOLOGIA E GESTÃO APLICADAS À REGENERAÇÃO TECIDUAL	LEILA BLANES	1898450330418640
33009015093P10	CIÊNCIAS, TECNOLOGIA E GESTÃO APLICADAS À REGENERAÇÃO TECIDUAL	LYDIA MASAKO FERREIRA	1619822351741819
33009015093P11	CIÊNCIAS, TECNOLOGIA E GESTÃO APLICADAS À REGENERAÇÃO TECIDUAL	RENATO SANTOS DE OLIVEIRA FILHO	3651646377594194
33009015038P1	CIRURGIA TRANSLACIONAL	ALFREDO GRAGNANI FILHO	5980775290384100
33009015038P2	CIRURGIA TRANSLACIONAL	DANIELA FRANCESCATO VEIGA	1695706360514926
33009015038P3	CIRURGIA TRANSLACIONAL	FABIO XERFAN NAHAS	3567288065904681
33009015038P4	CIRURGIA TRANSLACIONAL	FLAVIO FALOPPA	3695111273396745
33009015038P5	CIRURGIA TRANSLACIONAL	JOAO CARLOS BELLOTI	0981211406387862
33009015038P6	CIRURGIA TRANSLACIONAL	LYDIA MASAKO FERREIRA	1619822351741819
33009015038P7	CIRURGIA TRANSLACIONAL	MARCEL JUN SUGAWARA TAMAOKI	5982439031327655
33009015038P8	CIRURGIA TRANSLACIONAL	MAX DOMINGUES PEREIRA	9679136417299816
33009015038P9	CIRURGIA TRANSLACIONAL	MIGUEL SABINO NETO	2430172074494397
33009015038P10	CIRURGIA TRANSLACIONAL	MOISES COHEN	6174355233304675
33009015038P11	CIRURGIA TRANSLACIONAL	RENE JORGE ABDALLA	1751628419386085
33009015038P12	CIRURGIA TRANSLACIONAL	SILVIO EDUARDO DUAILIBI	3165201570670403
33009015014P5	MEDICINA (GINECOLOGIA)	AFONSO CELSO PINTO NAZARIO	0266384667983727
33009015014P6	MEDICINA (GINECOLOGIA)	CLELIA REJANE ANTONIO	6913838606609313

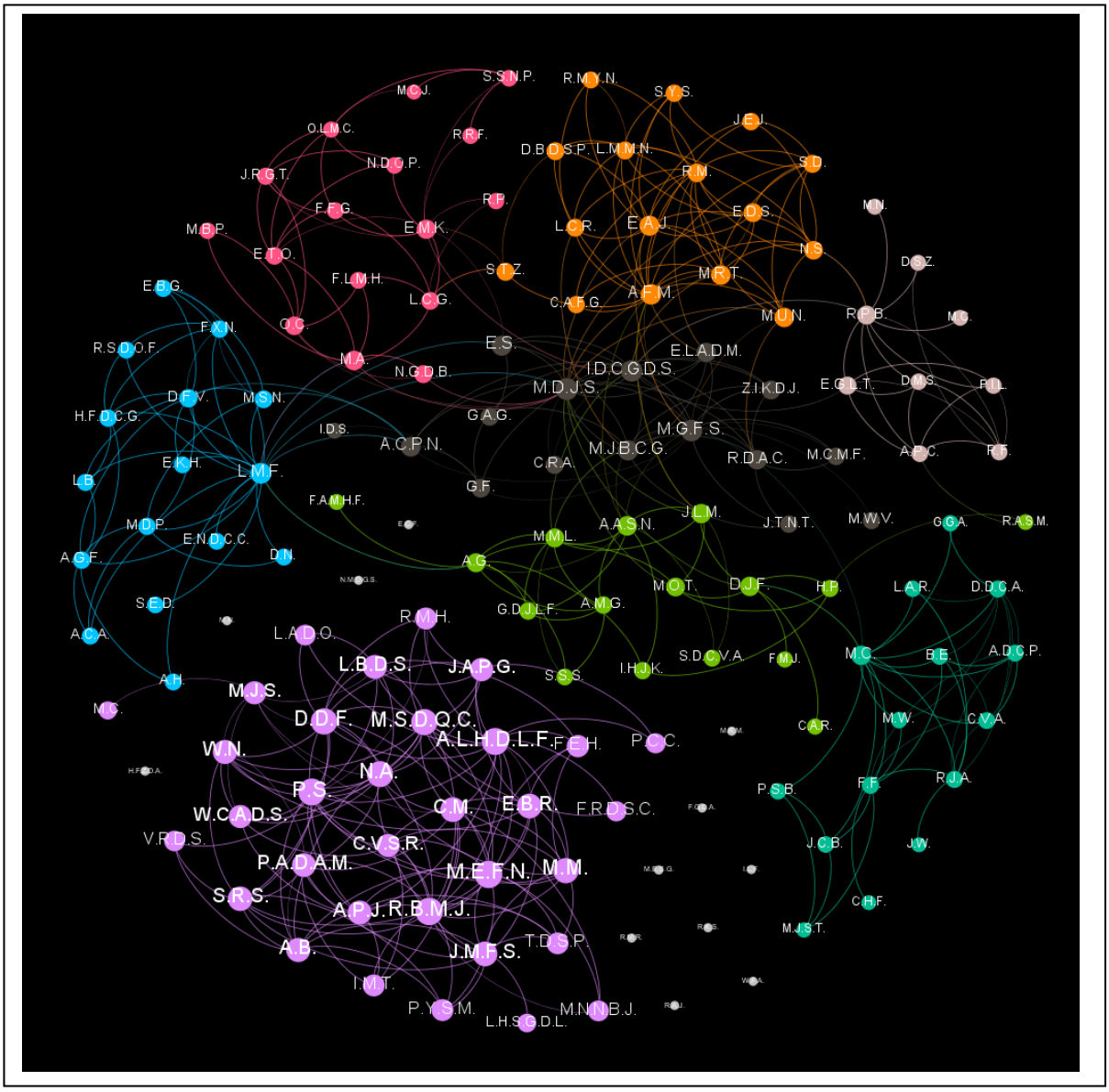
33009015014P7	MEDICINA (GINECOLOGIA)	EDUARDO LEME ALVES DA MOTTA	7532548942218501
33009015014P8	MEDICINA (GINECOLOGIA)	EDUARDO SCHOR	8854353153245040
33009015014P9	MEDICINA (GINECOLOGIA)	GIL FACINA	1029334251705417
33009015014P10	MEDICINA (GINECOLOGIA)	GIOVANA APARECIDA GONCALVES	5401245672797371
33009015014P11	MEDICINA (GINECOLOGIA)	ISMAEL DALE COTRIM GUERREIRO DA SILVA	7917312029683516
33009015014P12	MEDICINA (GINECOLOGIA)	IVALDO DA SILVA	9948402316193744
33009015014P13	MEDICINA (GINECOLOGIA)	JOSE TADEU NUNES TAMANINI	3679939808816276
33009015014P14	MEDICINA (GINECOLOGIA)	MANOEL JOAO BATISTA CASTELLO GIRAO	0973903299568770
33009015014P15	MEDICINA (GINECOLOGIA)	MANUEL DE JESUS SIMOES	5987164343458678
33009015014P16	MEDICINA (GINECOLOGIA)	MARAIR GRACIO FERREIRA SARTORI	2545470341657690
33009015014P17	MEDICINA (GINECOLOGIA)	MARCELO CUNIO MACHADO FONSECA	3881000445523759
33009015014P18	MEDICINA (GINECOLOGIA)	NEILA MARIA DE GOIS SPECK	8169544398769371
33009015014P19	MEDICINA (GINECOLOGIA)	RAFAEL MALAGOLI ROCHA	0319099107474192
33009015014P20	MEDICINA (GINECOLOGIA)	RODRIGO DE AQUINO CASTRO	6590913930590292
33009015014P21	MEDICINA (GINECOLOGIA)	ZSUZSANNA ILONA KATALIN DE JARMY	7368804318575164
33009015013P9	MEDICINA (OBSTETRÍCIA)	ANTONIO FERNANDES MORON	0197731060424158
33009015013P10	MEDICINA (OBSTETRÍCIA)	CRISTINA APARECIDA FALBO GUAZZELLI	4413607144407436
33009015013P11	MEDICINA (OBSTETRÍCIA)	DAVID BAPTISTA DA SILVA PARES	6323380679327937
33009015013P12	MEDICINA (OBSTETRÍCIA)	EDUARDO DE SOUZA	5771296988055873
33009015013P13	MEDICINA (OBSTETRÍCIA)	EDWARD ARAUJO JUNIOR	5590809884662013
33009015013P14	MEDICINA (OBSTETRÍCIA)	JULIO ELITO JUNIOR	3134344285801202
33009015013P15	MEDICINA (OBSTETRÍCIA)	LILIAM CRISTINE ROLO	7732742031806411
33009015013P16	MEDICINA (OBSTETRÍCIA)	LUCIANO MARCONDES MACHADO NARDOZZA	1184942167958634
33009015013P17	MEDICINA (OBSTETRÍCIA)	MARIA REGINA TORLONI	5661395483781554
33009015013P18	MEDICINA (OBSTETRÍCIA)	MARY UCHIYAMA NAKAMURA	4320107502315102
33009015013P19	MEDICINA (OBSTETRÍCIA)	NELSON SASS	6079546404174722
33009015013P20	MEDICINA (OBSTETRÍCIA)	ROSELI MIEKO YAMAMOTO NOMURA	1256048327375313
33009015013P21	MEDICINA (OBSTETRÍCIA)	ROSIANE MATTAR	1993353561775961
33009015013P22	MEDICINA (OBSTETRÍCIA)	SILVIA DAHER	5938358901097469
33009015013P23		SUE YAZAKI SUN	3253295555494164
33009015018P0	MEDICINA (OTORRINOLARINGOLOGIA)	EDUARDO MACOTO KOSUGI	9771826548166046

33009015018P1	MEDICINA (OTORRINOLARINGOLOGIA)	EKTOR TSUNEO ONISHI	9383669632593200
33009015018P2	MEDICINA (OTORRINOLARINGOLOGIA)	FERNANDA LOUISE MARTINHO HADDAD	2110917250638917
33009015018P3	MEDICINA (OTORRINOLARINGOLOGIA)	FERNANDO FREITAS GANANCA	0831776469482702
33009015018P4	MEDICINA (OTORRINOLARINGOLOGIA)	JOSE RICARDO GURGEL TESTA	1154965263654209
33009015018P5	MEDICINA (OTORRINOLARINGOLOGIA)	LUIS CARLOS GREGORIO	3121718741179338
33009015018P6	MEDICINA (OTORRINOLARINGOLOGIA)	MARCIO ABRAHAO	4440305851848835
33009015018P7	MEDICINA (OTORRINOLARINGOLOGIA)	MARCOS BANDIERA PAIVA	2154594986091571
33009015018P8	MEDICINA (OTORRINOLARINGOLOGIA)	MARIO CAPPELLETTE JUNIOR	3772804052798387
33009015018P9	MEDICINA (OTORRINOLARINGOLOGIA)	MAURO WALTER VAISBERG	3009013165831084
33009015018P10	MEDICINA (OTORRINOLARINGOLOGIA)	NOEMI GRIGOLETTO DE BIASE	3156326658988323
33009015018P11	MEDICINA (OTORRINOLARINGOLOGIA)	NORMA DE OLIVEIRA PENIDO	7060786297081212
33009015018P12	MEDICINA (OTORRINOLARINGOLOGIA)	ONIVALDO CERVANTES	2752448898797822
33009015018P13	MEDICINA (OTORRINOLARINGOLOGIA)	OSWALDO LAERCIO MENDONCA CRUZ	3152241414526004
33009015018P14	MEDICINA (OTORRINOLARINGOLOGIA)	REGINALDO RAIMUNDO FUJITA	1780341325141181
33009015018P15	MEDICINA (OTORRINOLARINGOLOGIA)	ROGERIO PEZATO	8850675385685321
33009015018P16	MEDICINA (OTORRINOLARINGOLOGIA)	SAMUEL TAU ZYMBERG	9399440722396953
33009015018P17	MEDICINA (OTORRINOLARINGOLOGIA)	SHIRLEY SHIZUE NAGATA PIGNATARI	4416616059943202
33009015021P1	MEDICINA (UROLOGIA)	AGNALDO PEREIRA CEDENHO	1386922092780490
33009015021P2	MEDICINA (UROLOGIA)	CASSIO ANDREONI RIBEIRO	2915020488175752
33009015021P3	MEDICINA (UROLOGIA)	DANIEL SUSLIK ZYLBERSZTEJN	2798788209202563
33009015021P4	MEDICINA (UROLOGIA)	DEBORAH MONTAGNINI SPAINE	1321284998759806
33009015021P5	MEDICINA (UROLOGIA)	EDSON GUIMARAES LO TURCO	0251394799200508
33009015021P6	MEDICINA (UROLOGIA)	FERNANDO GONCALVES DE ALMEIDA	9336208077764596
33009015021P7	MEDICINA (UROLOGIA)	HATYLAS FELYPE ZANETI DE AZEVEDO	7580295280253981
33009015021P8	MEDICINA (UROLOGIA)	MARCILIO NICHI	7054360633543023
33009015021P9	MEDICINA (UROLOGIA)	MARIANA CAMARGO	0574614880460399
33009015021P10	MEDICINA (UROLOGIA)	PAULA INTASQUI LOPES	5188240479960852
33009015021P11	MEDICINA (UROLOGIA)	RENATO FRAIETTA	1545035937368744
33009015021P12	MEDICINA (UROLOGIA)	RICARDO PIMENTA BERTOLLA	8479803539567479

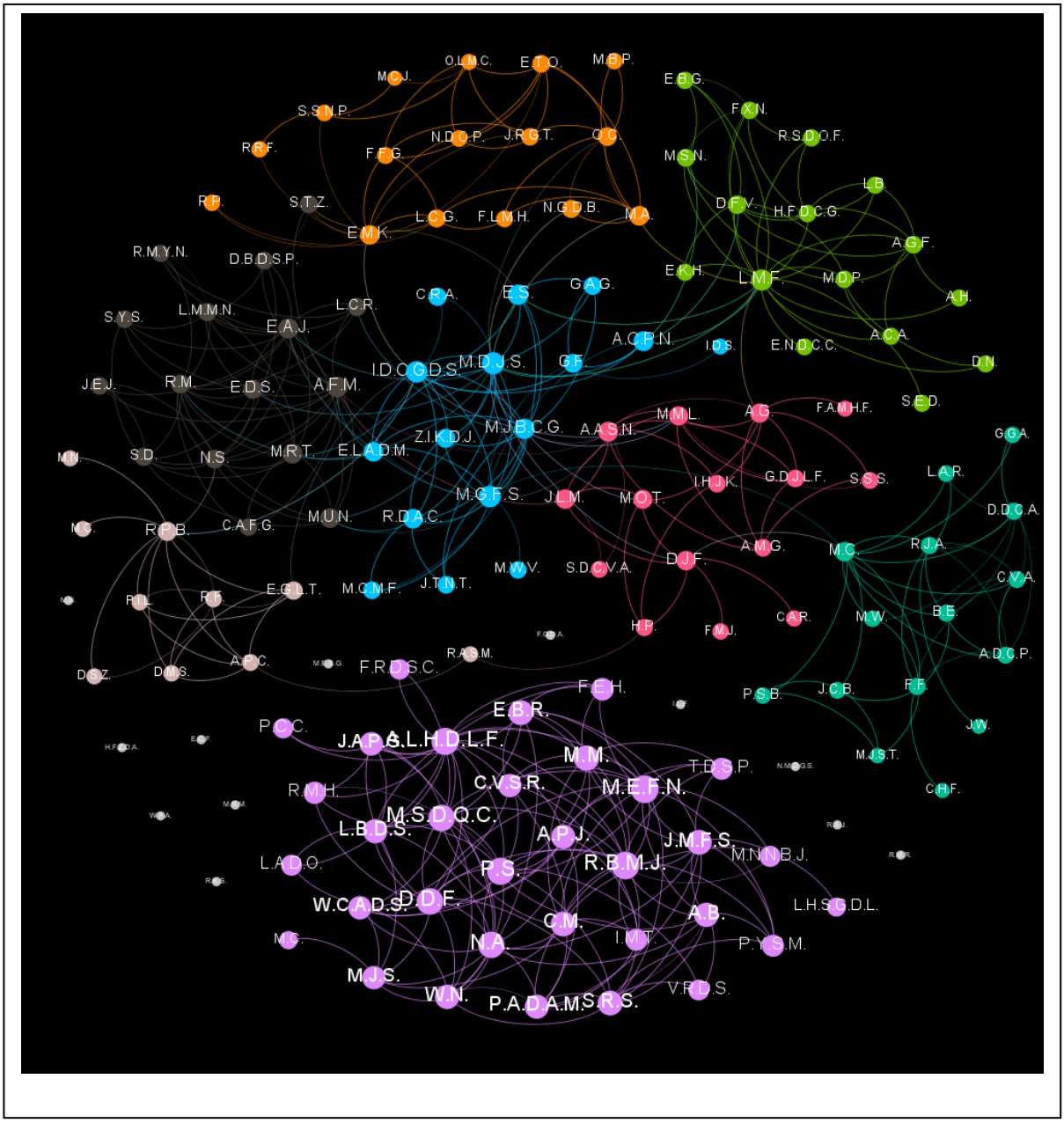
33009015021P13	MEDICINA (UROLOGIA)	ROBERTO ANDRE SOLER MESQUITA	9038872306641159
33009015021P14	MEDICINA (UROLOGIA)	WILSON FERREIRA AGUIAR	1555985311877836
33009015024P0	Oftalmologia e Ciências Visuais	ADRIANA BEREZOVSKY	1336685192142760
33009015024P1	Oftalmologia e Ciências Visuais	ANA LUISA HOFLING DE LIMA FARAH	7050225867972978
33009015024P2	Oftalmologia e Ciências Visuais	AUGUSTO PARANHOS JUNIOR	8476823547757955
33009015024P3	Oftalmologia e Ciências Visuais	CAIO VINICIUS SAITO REGATIERI	3274977342997227
33009015024P4	Oftalmologia e Ciências Visuais	CRISTINA MUCCIOLI	4512517971941945
33009015024P5	Oftalmologia e Ciências Visuais	DENISE DE FREITAS	4036480252471491
33009015024P6	Oftalmologia e Ciências Visuais	EDUARDO BUCHELE RODRIGUES	4226917385383502
33009015024P7	Oftalmologia e Ciências Visuais	FABIO RAMOS DE SOUZA CARVALHO	1910912718767159
33009015024P8	Oftalmologia e Ciências Visuais	IVAN MAYNART TAVARES	9220084935194992
33009015024P9	Oftalmologia e Ciências Visuais	JOSE ALVARO PEREIRA GOMES	2506247984223015
33009015024P10	Oftalmologia e Ciências Visuais	JULIANA MARIA FERRAZ SALLUM	2233267084488852
33009015024P11	Oftalmologia e Ciências Visuais	LAURO AUGUSTO DE OLIVEIRA	5591822363508144
33009015024P12	Oftalmologia e Ciências Visuais	LUCIENE BARBOSA DE SOUSA	2367491641205774
33009015024P13	Oftalmologia e Ciências Visuais	MAURICIO MAIA	6377105744231862
33009015024P14	Oftalmologia e Ciências Visuais	MAURO SILVEIRA DE QUEIROZ CAMPOS	8668472375424523
33009015024P15	Oftalmologia e Ciências Visuais	MICHEL EID FARAH NETO	1907009763960478
33009015024P16	Oftalmologia e Ciências Visuais	MIGUEL NOEL NASCENTES BURNIER JUNIOR	4487521900423032
33009015024P17	Oftalmologia e Ciências Visuais	NORMA ALLEMANN	0956596522261307
33009015024P18	Oftalmologia e Ciências Visuais	PAULO AUGUSTO DE ARRUDA MELLO	9184537433303452
33009015024P19	Oftalmologia e Ciências Visuais	PAULO SCHOR	3542867700396961
33009015024P20	Oftalmologia e Ciências Visuais	RENATO AMBROSIO JUNIOR	1789497818458326
33009015024P21	Oftalmologia e Ciências Visuais	RUBENS BELFORT MATTOS JUNIOR	4270399167335564
33009015024P22	Oftalmologia e Ciências Visuais	SOLANGE RIOS SALOMAO	0725782105811003
33009015024P23	Oftalmologia e Ciências Visuais	TIAGO DOS SANTOS PRATA	4375022069702250
33009015024P24	Oftalmologia e Ciências Visuais	WALLACE CHAMON ALVES DE SIQUEIRA	3165995344927892
33009015024P25	Oftalmologia e Ciências Visuais	WALTON NOSE	0107737030151731
33009015082P0	Tecnologia, Gestão e Saúde Ocular	ADRIANA BEREZOVSKY	1336685192142760
33009015082P1	Tecnologia, Gestão e Saúde Ocular	CRISTINA MUCCIOLI	4512517971941945
33009015082P2	Tecnologia, Gestão e Saúde Ocular	DENISE DE FREITAS	4036480252471491
33009015082P3	Tecnologia, Gestão e Saúde Ocular	ELIANA CHAVES FERRETTI	6560097254021874

33009015082P4	Tecnologia, Gestão e Saúde Ocular	FABIO RAMOS DE SOUZA CARVALHO	1910912718767159
33009015082P5	Tecnologia, Gestão e Saúde Ocular	FLAVIO EDUARDO HIRAI	6575096591259140
33009015082P6	Tecnologia, Gestão e Saúde Ocular	IVAN MAYNART TAVARES	9220084935194992
33009015082P7	Tecnologia, Gestão e Saúde Ocular	JOSE ALVARO PEREIRA GOMES	2506247984223015
33009015082P8	Tecnologia, Gestão e Saúde Ocular	LUCIENE BARBOSA DE SOUSA	2367491641205774
33009015082P9	Tecnologia, Gestão e Saúde Ocular	LUIZ HENRIQUE SOARES GONCALVES DE LIMA	0399713306487727
33009015082P10	Tecnologia, Gestão e Saúde Ocular	MARCELO CONTE	8945297747305462
33009015082P11	Tecnologia, Gestão e Saúde Ocular	MARCIA ROCHA MONTEIRO	4930117991668673
33009015082P12	Tecnologia, Gestão e Saúde Ocular	MARIA ELISABETE SALVADOR GRAZIOSI	2940944860729134
33009015082P13	Tecnologia, Gestão e Saúde Ocular	MARINHO JORGE SCARPI	4849663856118153
33009015082P14	Tecnologia, Gestão e Saúde Ocular	MARTINA NAVARRO	3531342066691171
33009015082P16	Tecnologia, Gestão e Saúde Ocular	NORMA ALLEMANN	0956596522261307
33009015082P17	Tecnologia, Gestão e Saúde Ocular	PAULA YURI SACAI MUNHOZ	6877836442964718
33009015082P18	Tecnologia, Gestão e Saúde Ocular	PRISCILA CARDOSO CRISTOVAM	4581855390314828
33009015082P19	Tecnologia, Gestão e Saúde Ocular	ROSSEN MIHAYLOV HAZARBASSANOV	8094321275686016
33009015082P20	Tecnologia, Gestão e Saúde Ocular	RUBENS BELFORT MATTOS JUNIOR	4270399167335564
33009015082P21	Tecnologia, Gestão e Saúde Ocular	VAGNER ROGERIO DOS SANTOS	0921491281575273

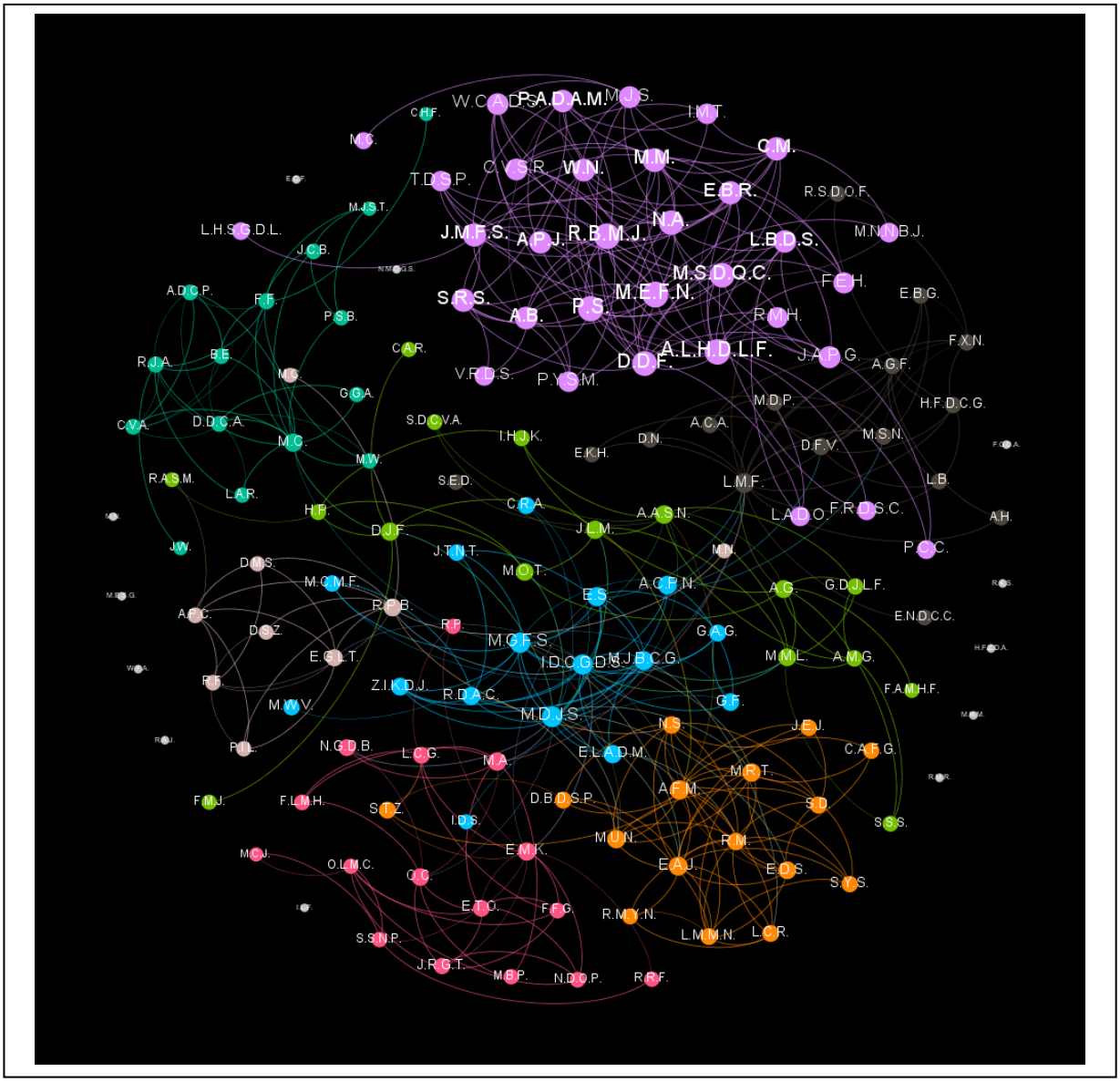
Anexo 3 | Redes de colaboração por Programa



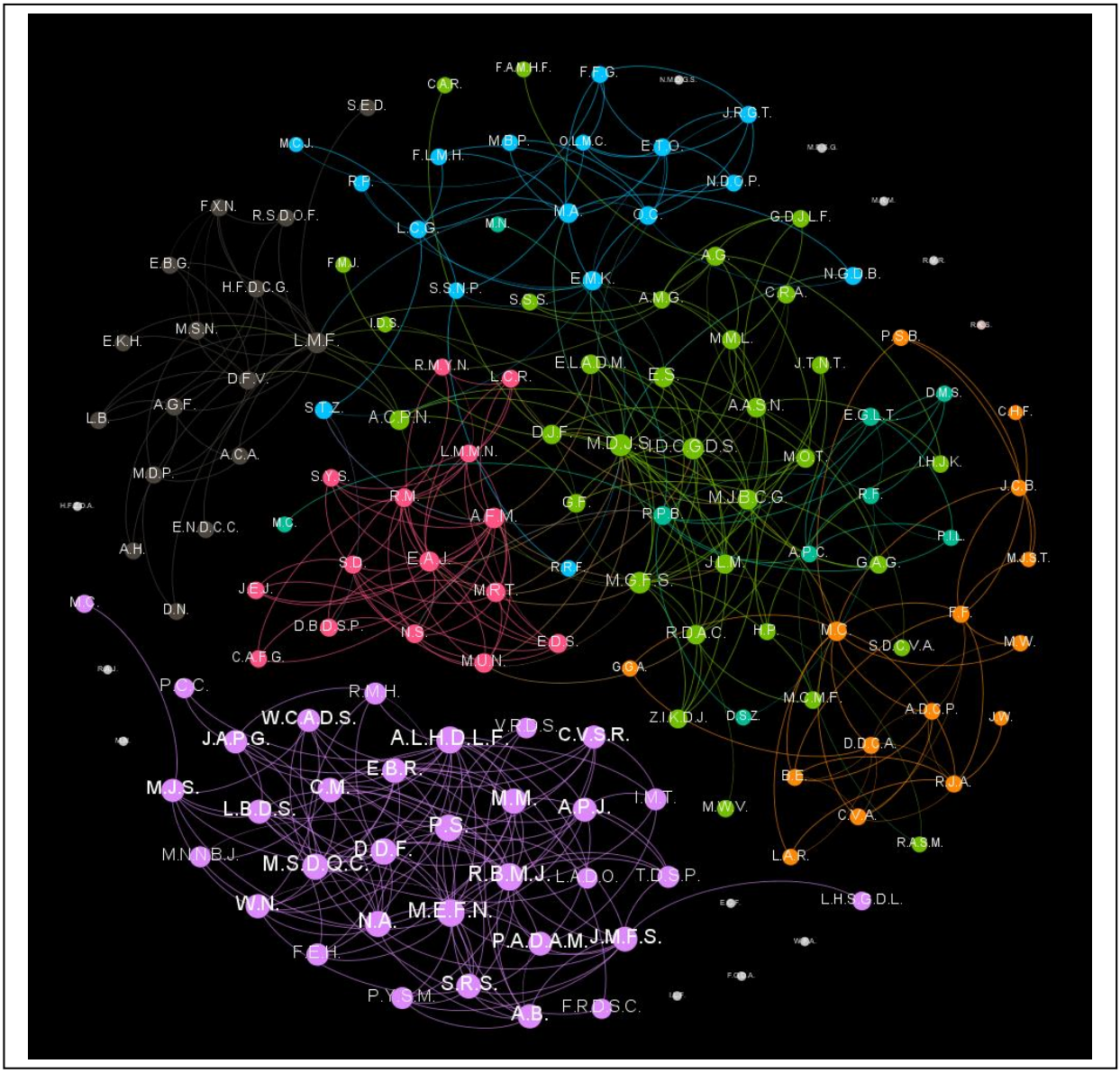
Rede de colaboração da Medicina III 2010-2012 – Unifesp



Rede de colaboração da Medicina III 2010-2018 – Unifesp

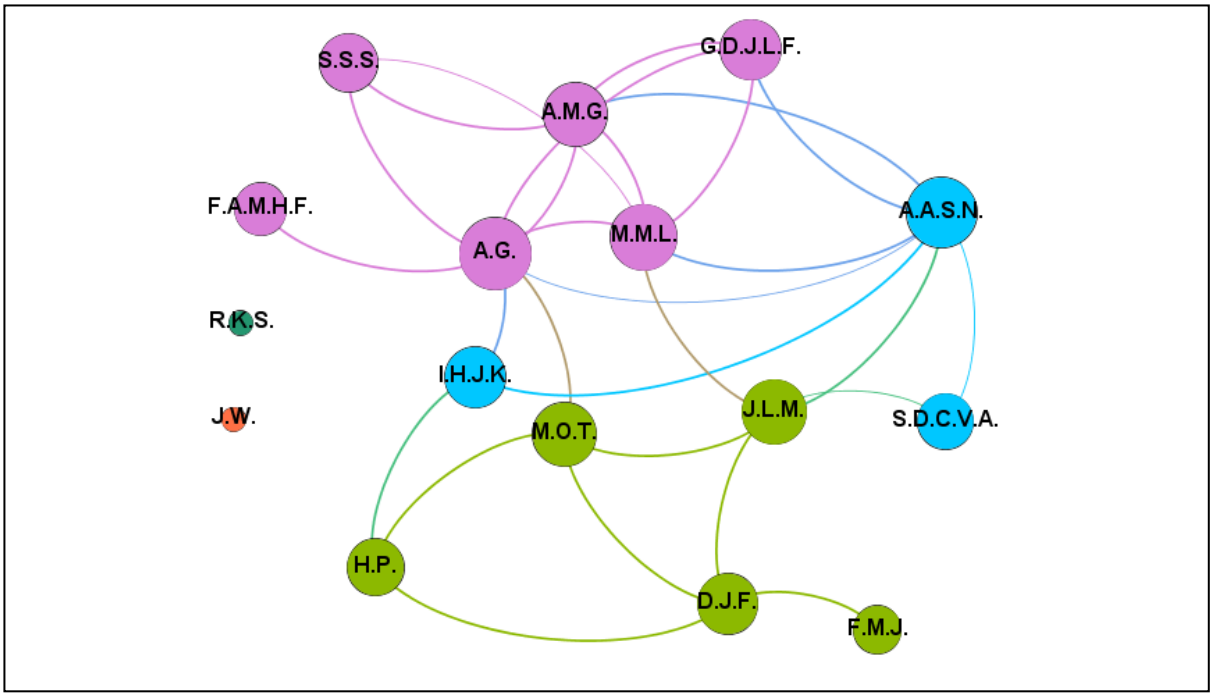


Rede de colaboração da Medicina III 2013-2016– Unifesp

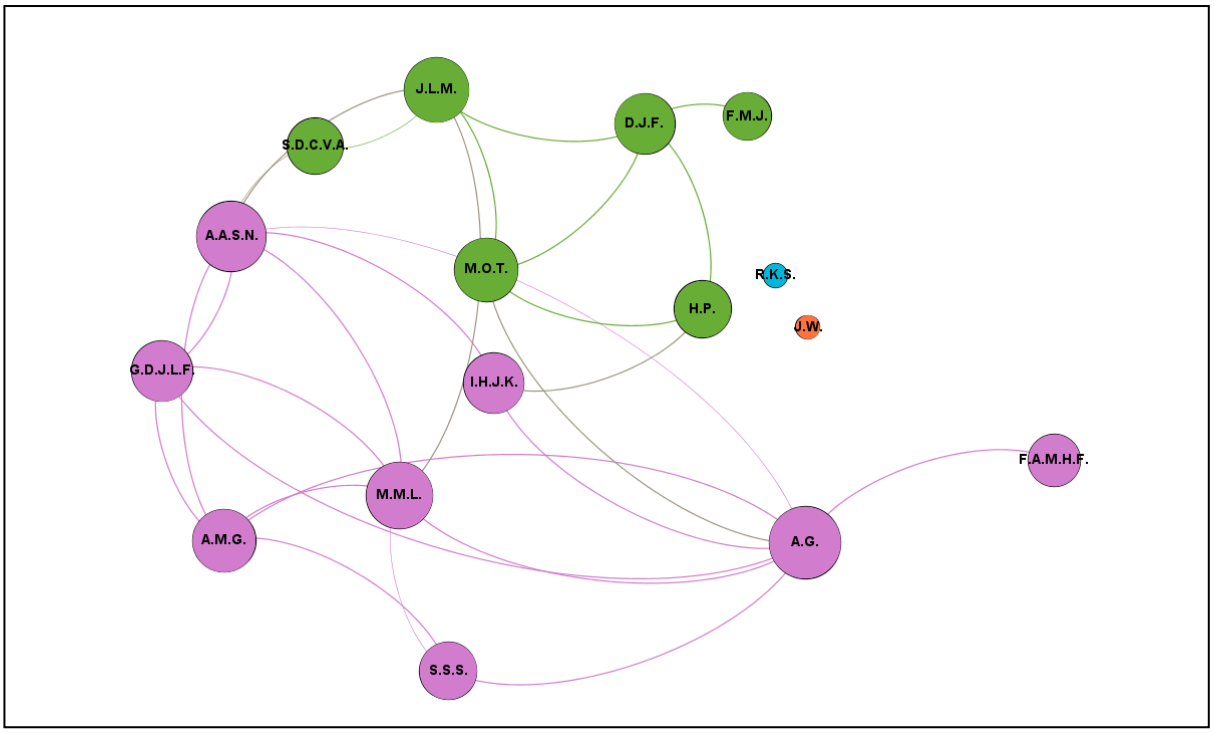


Rede de colaboração da Medicina III 2017-2018– Unifesp

Ciências e Cirurgia interdisciplinar

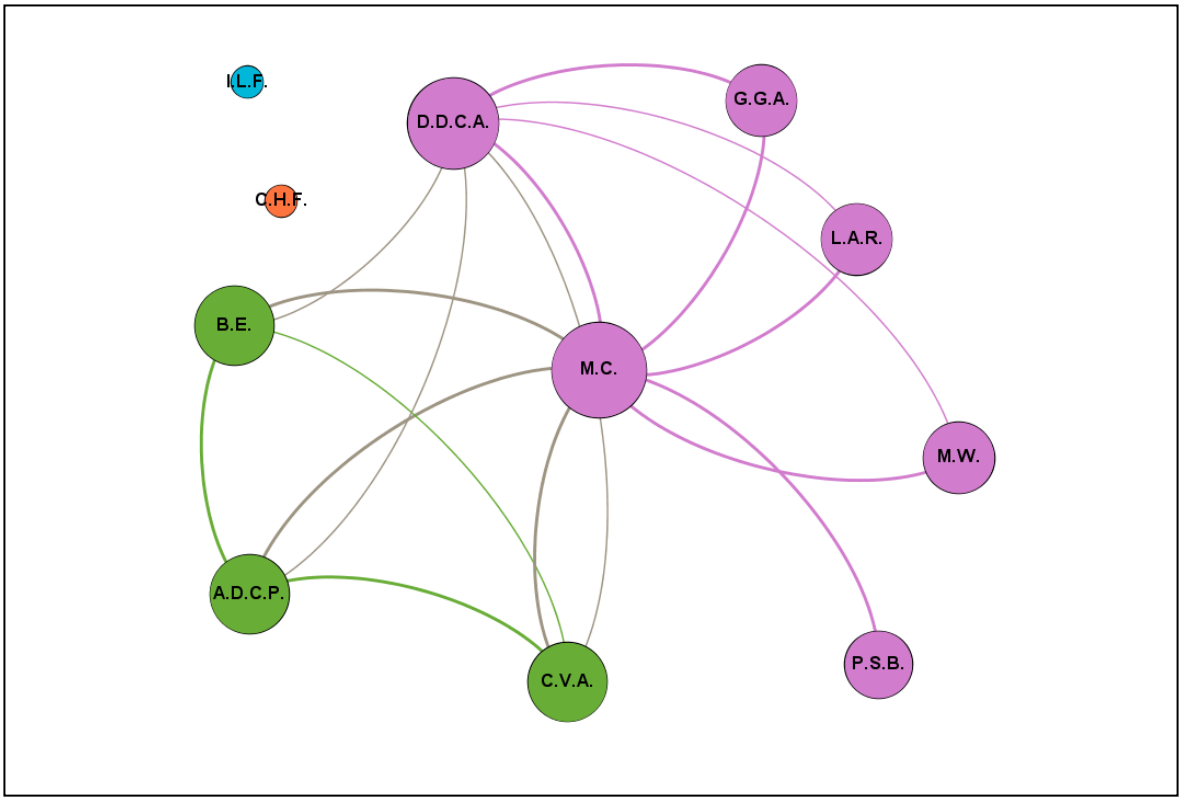


Ciências e Cirurgia interdisciplinar (2010-2012)

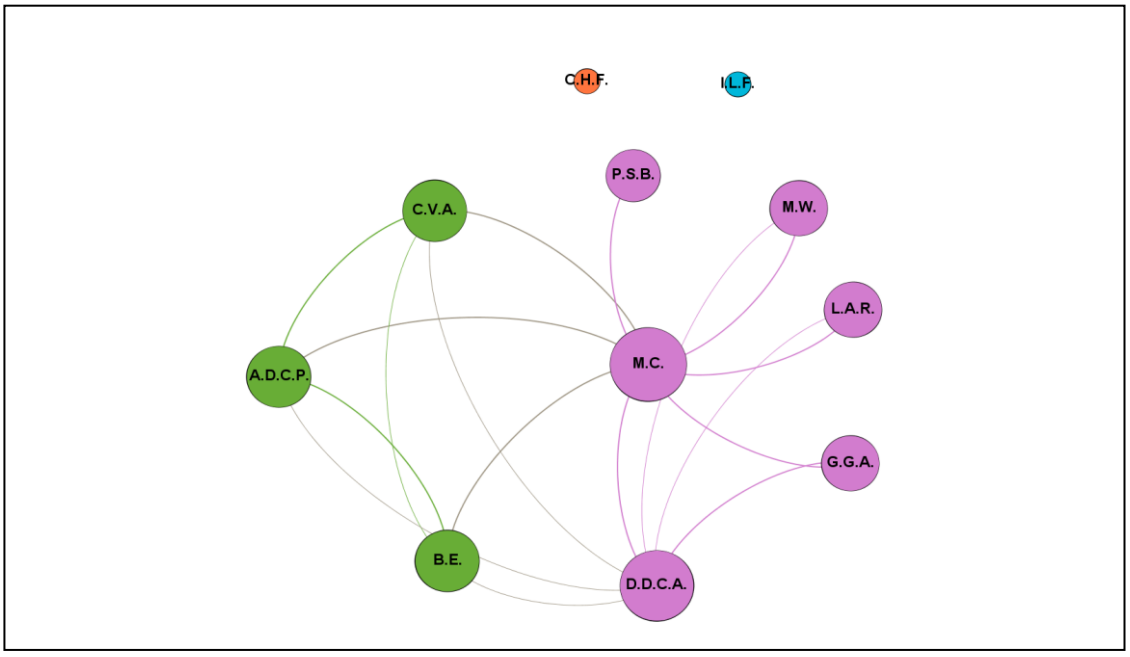


Ciências e Cirurgia interdisciplinar (2010-2018)

Ciência da Saúde Aplicada ao Esporte

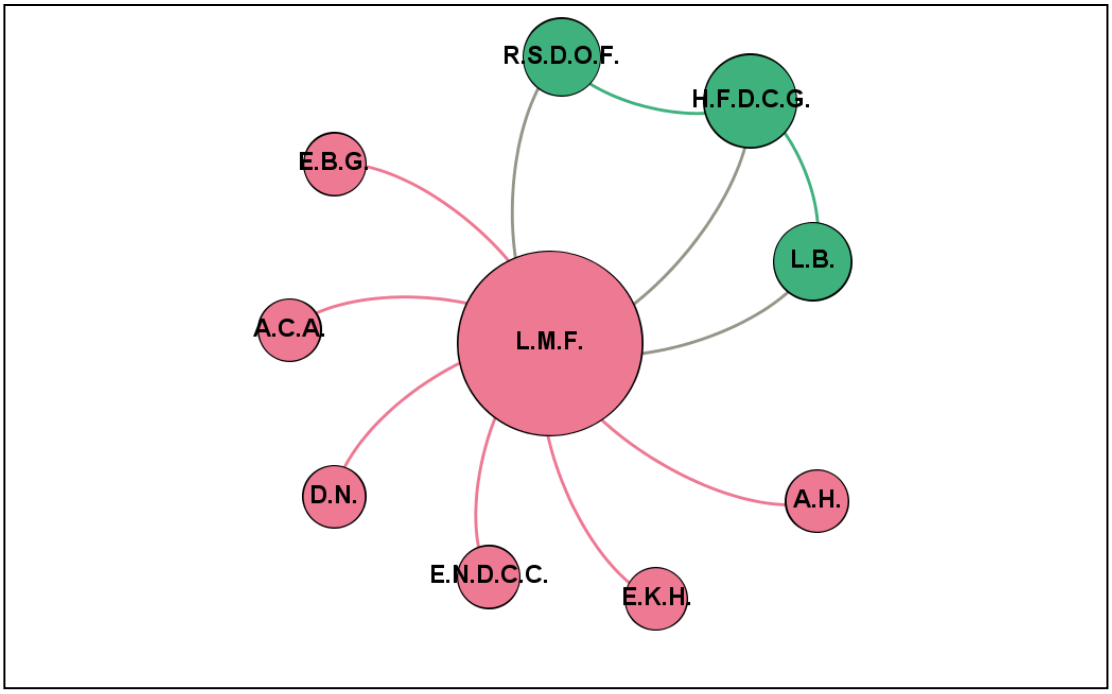


Ciência da Saúde Aplicada ao Esporte (2010-2012)

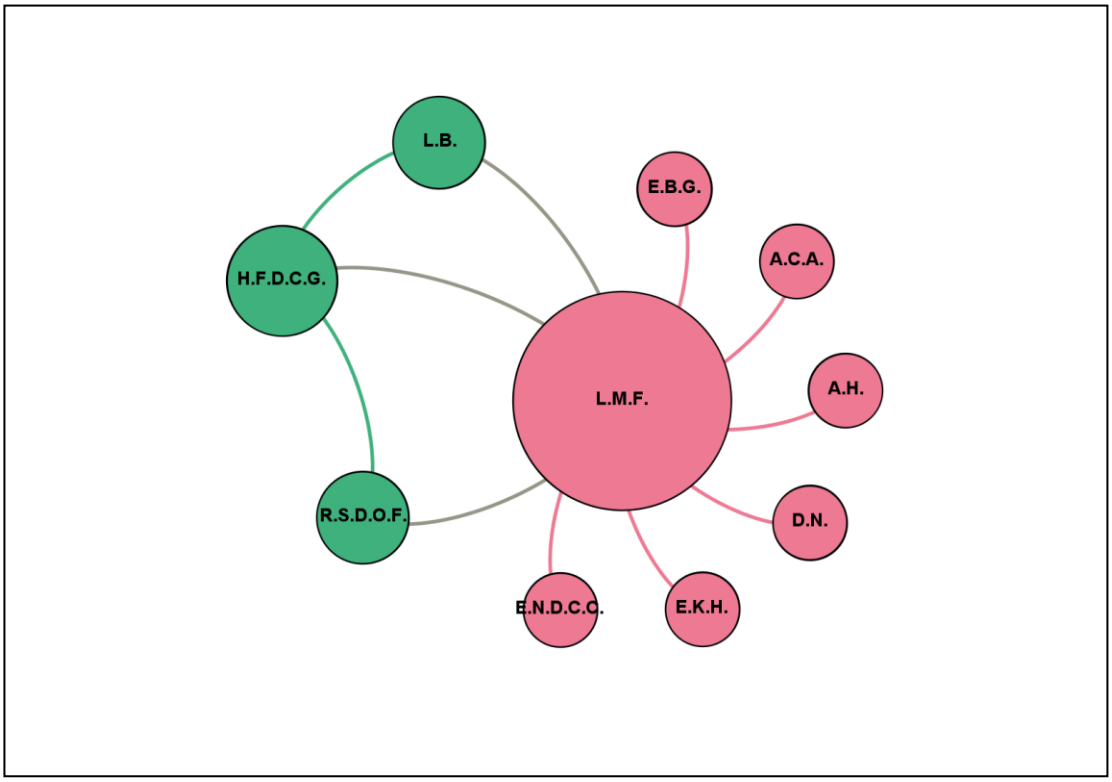


Ciência da Saúde Aplicada ao Esporte (2010-2018)

Ciências Tecnologia e Gestão Aplicada a Regeneração Tecidual

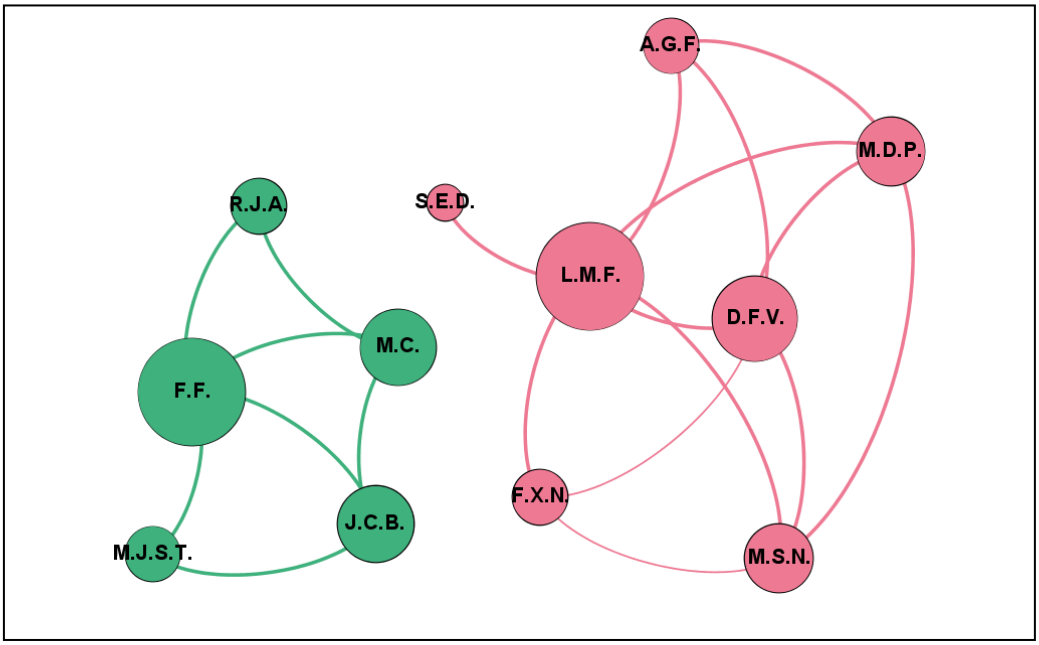


Ciências Tecnologia e Gestão Aplicada a Regeneração Tecidual (2010-2012)

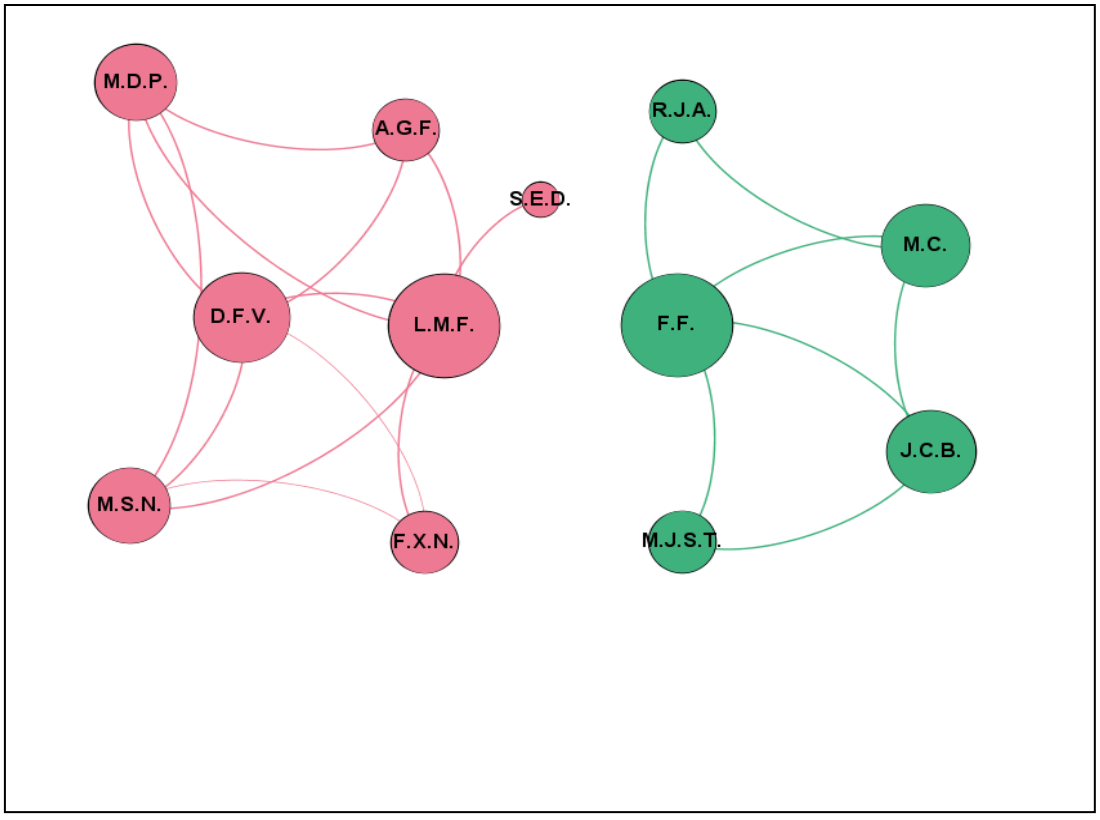


Ciências Tecnologia e Gestão Aplicada a Regeneração Tecidual (2010-2018)

Cirurgia Translacional

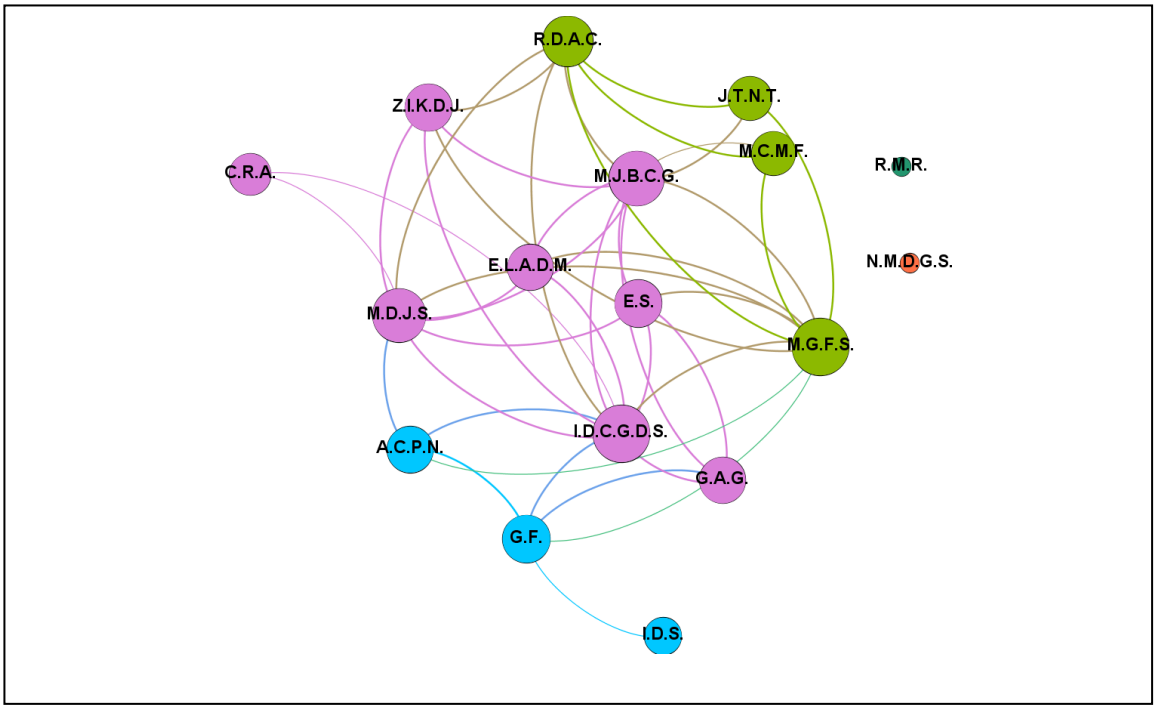


Cirurgia Translacional (2010-2012)

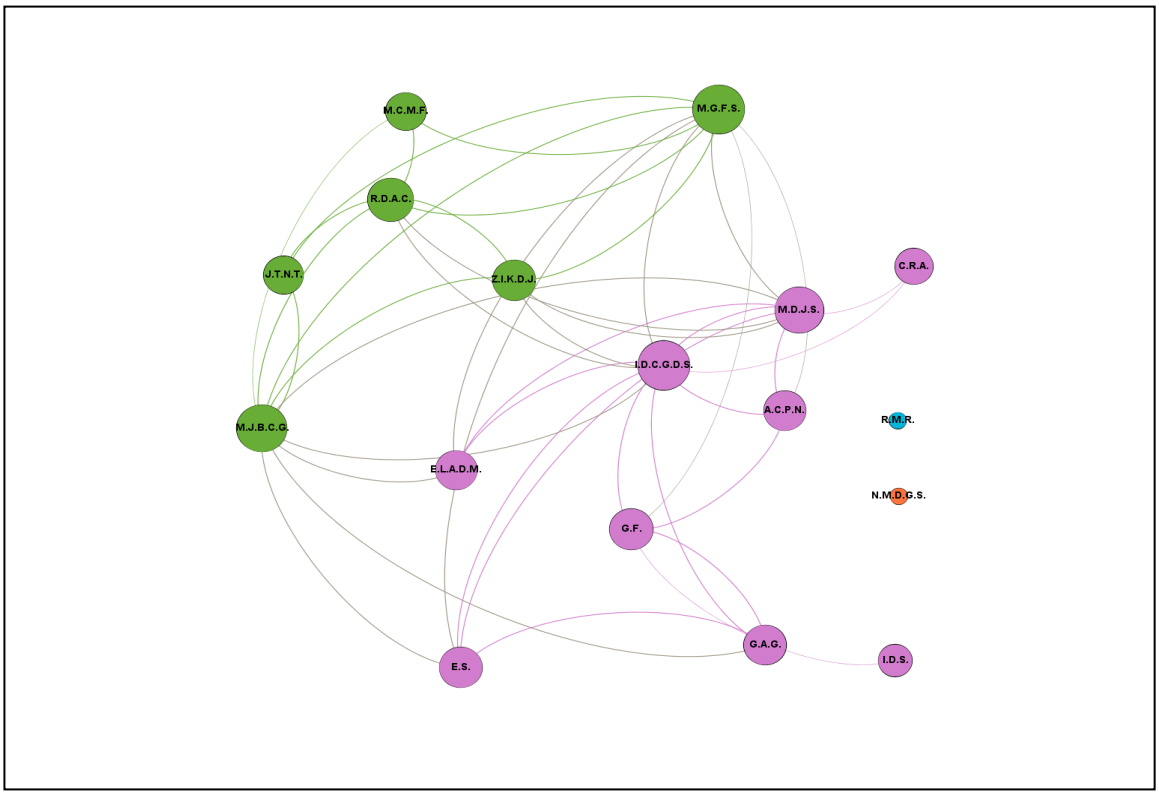


Cirurgia Translacional (2010-2018)

Medicina Ginecologia

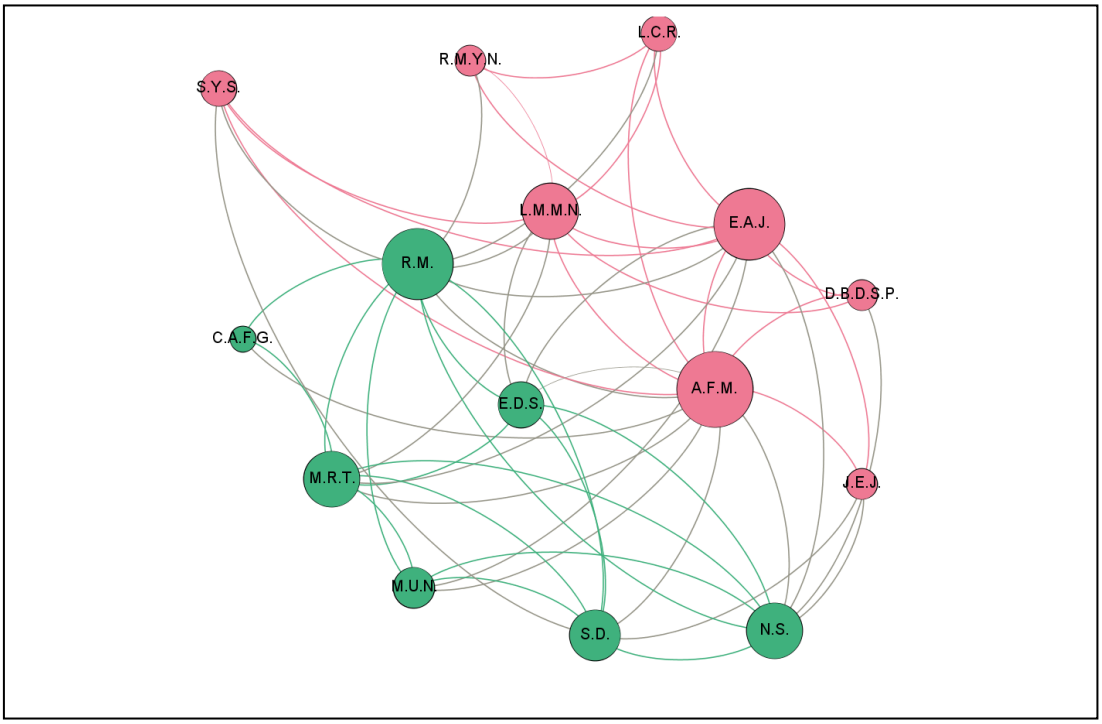


Medicina Ginecologia (2010-2012)

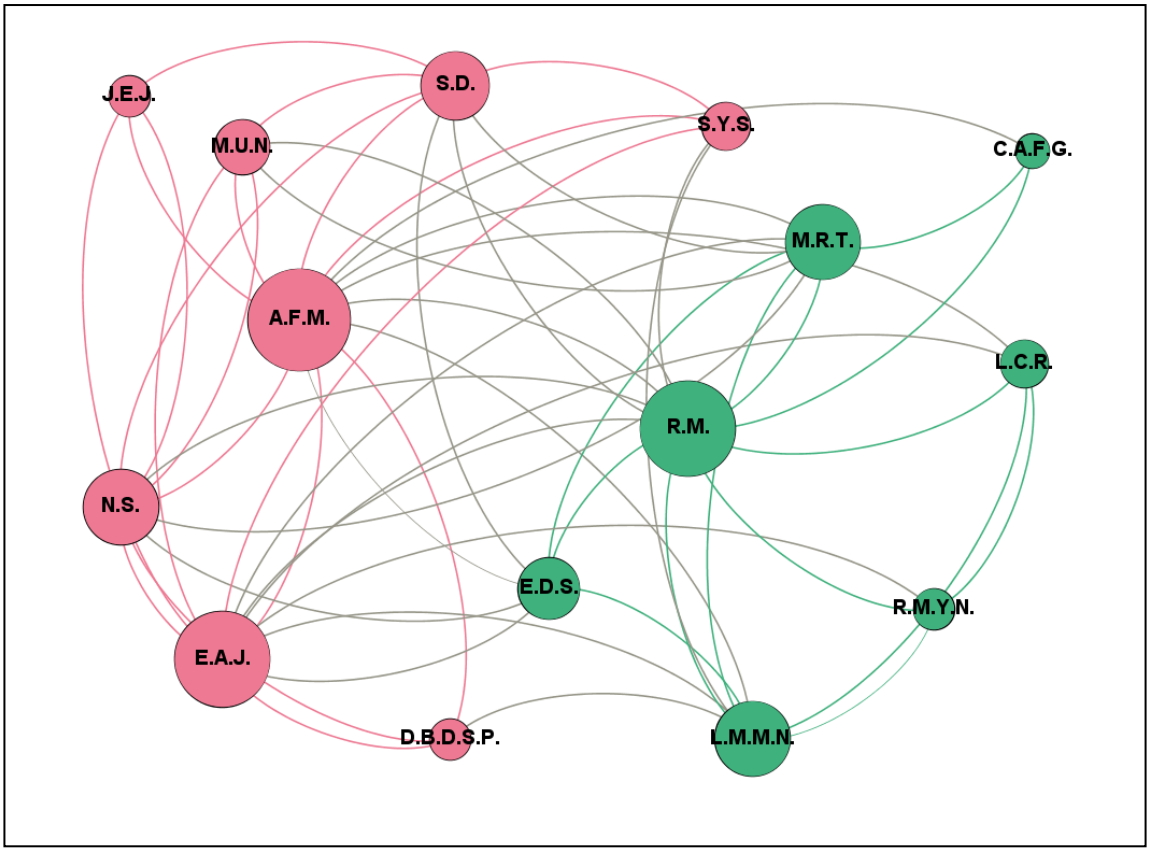


Medicina Ginecologia (2010-2018)

Medicina Obstetrícia

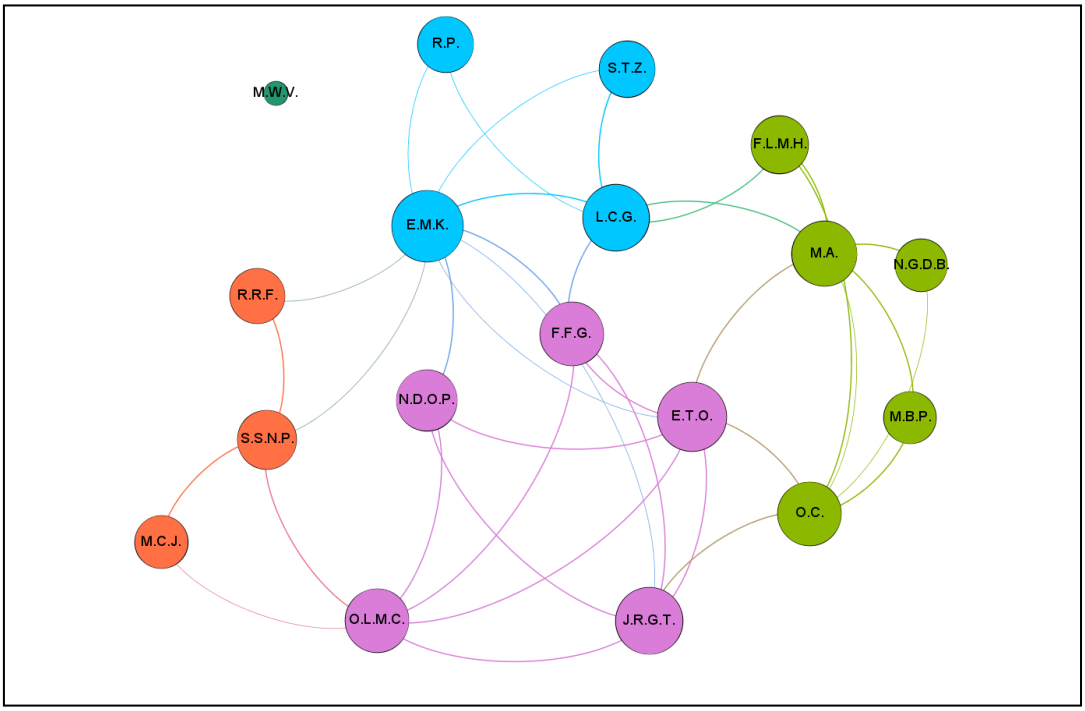


Medicina Obstetrícia (2010-2012)

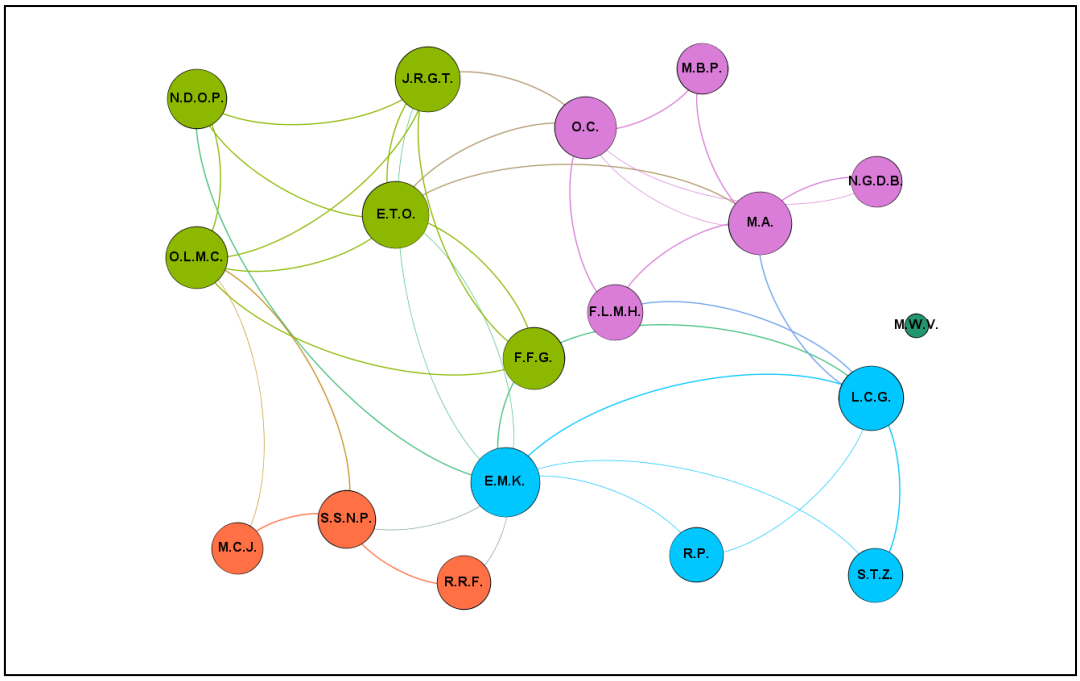


Medicina Obstetrícia (2010-2018)

Medicina Otorrinolaringologia

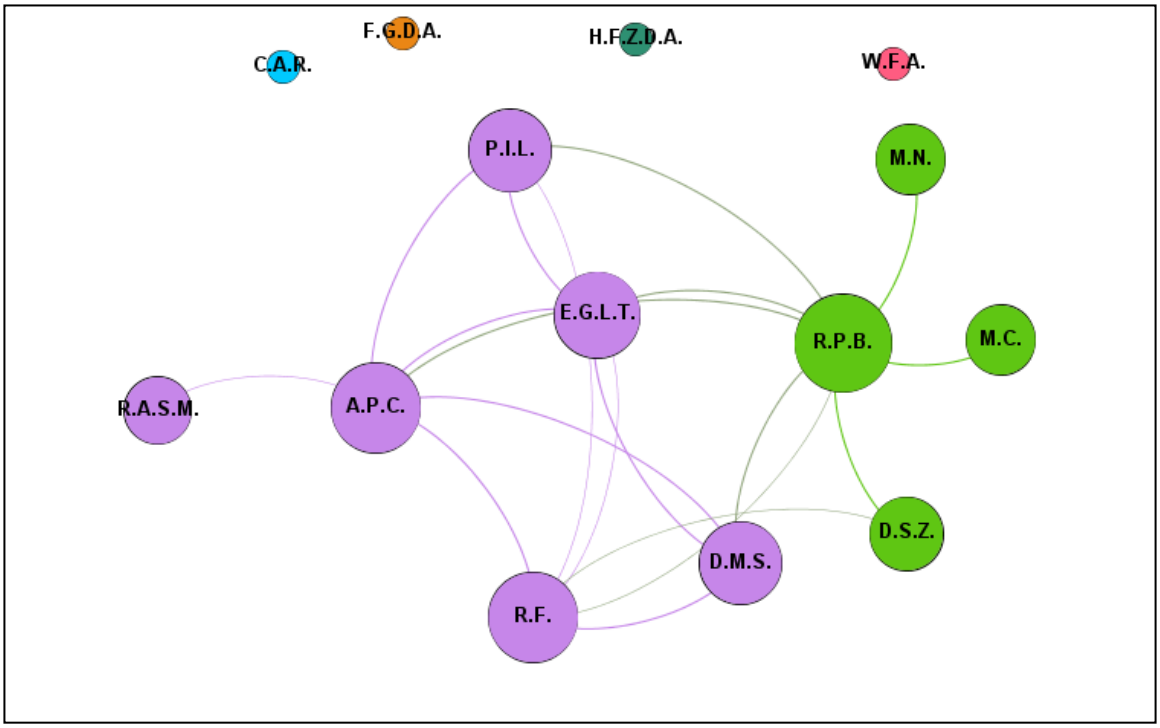


Medicina Otorrinolaringologia (2010-2012)

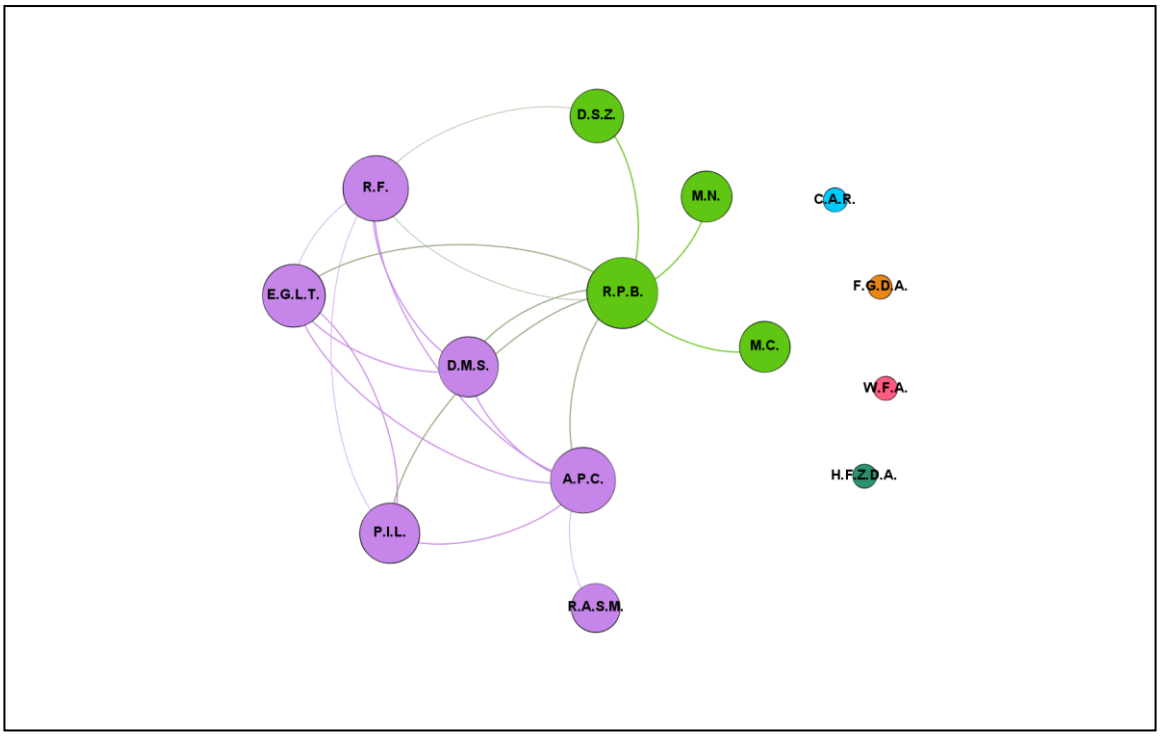


Medicina Otorrinolaringologia (2010-2018)

Medicina Urologia

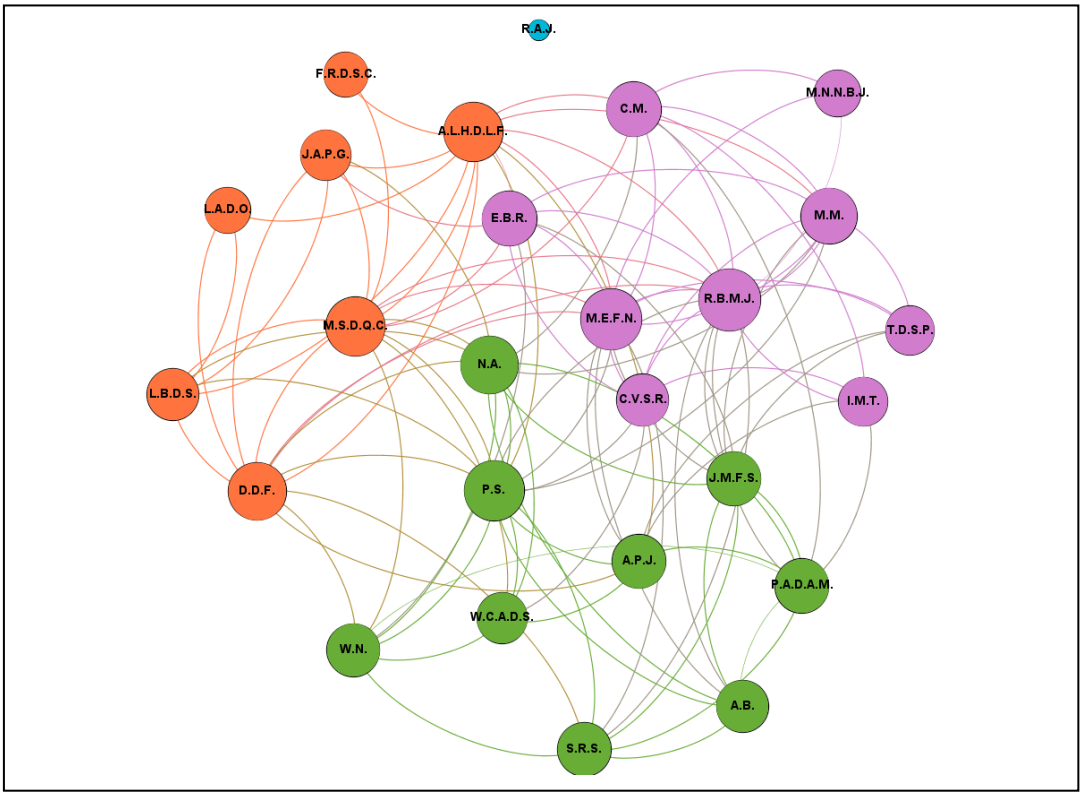


Medicina Urologia (2010-2012)

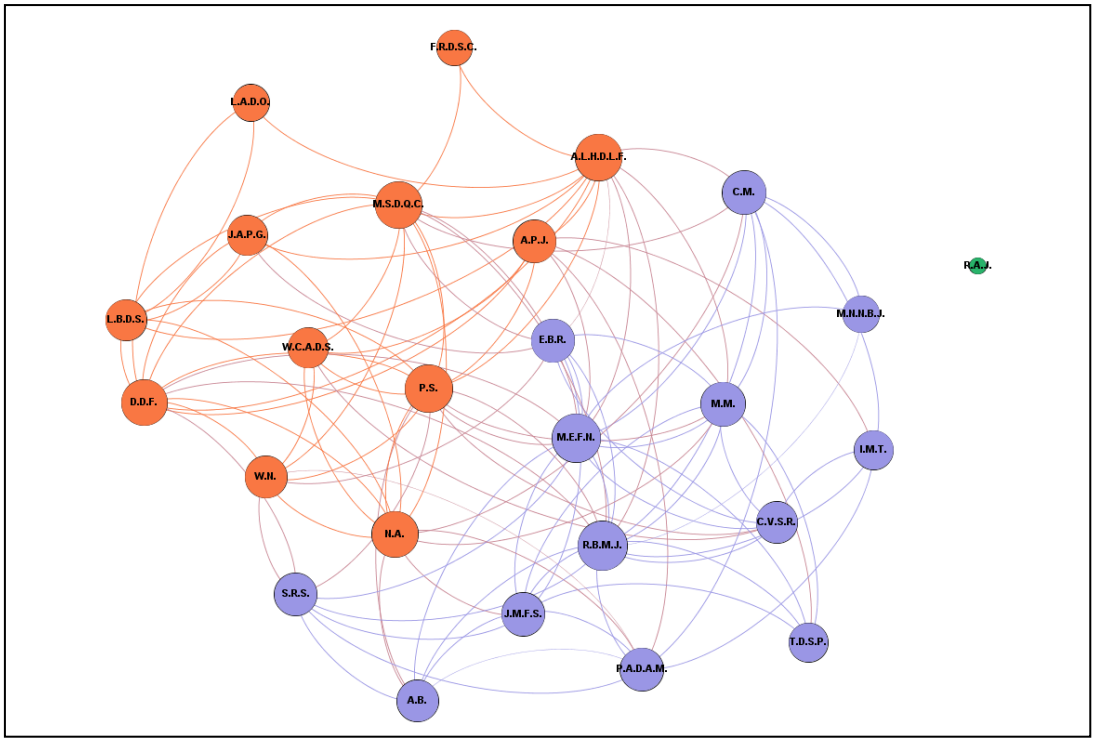


Medicina Urologia (2010-2018)

Oftalmologia Ciências Visuais

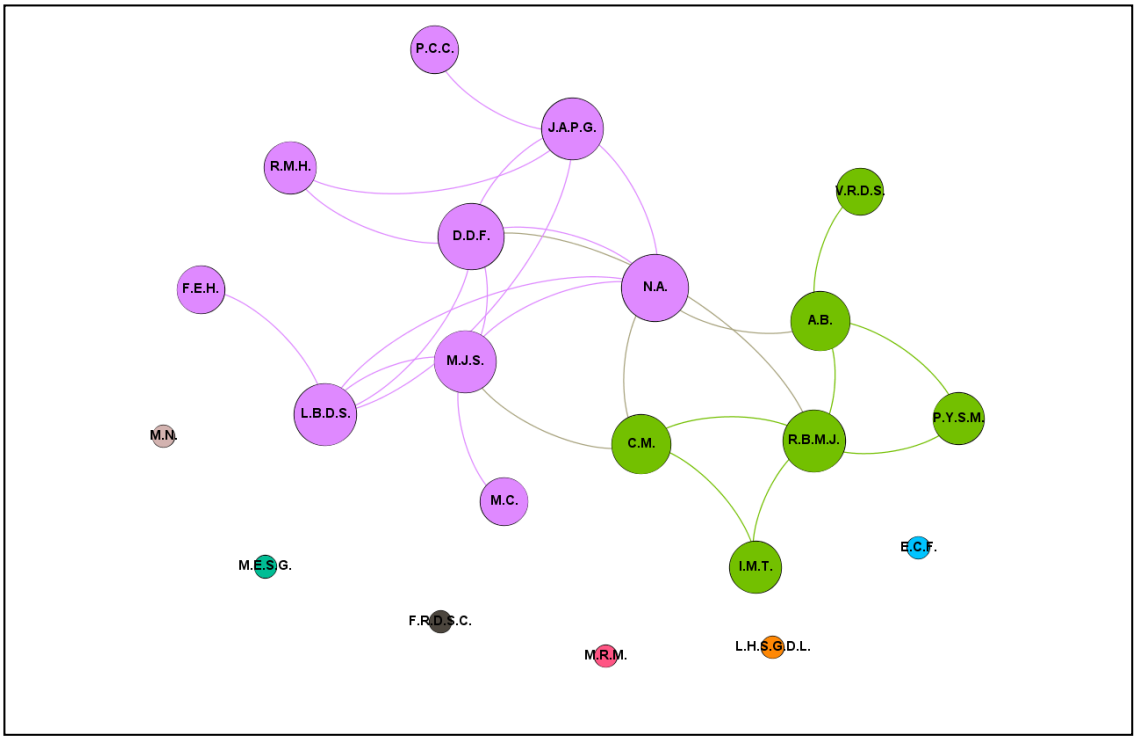


Oftalmologia Ciências Visuais (2010-2012)

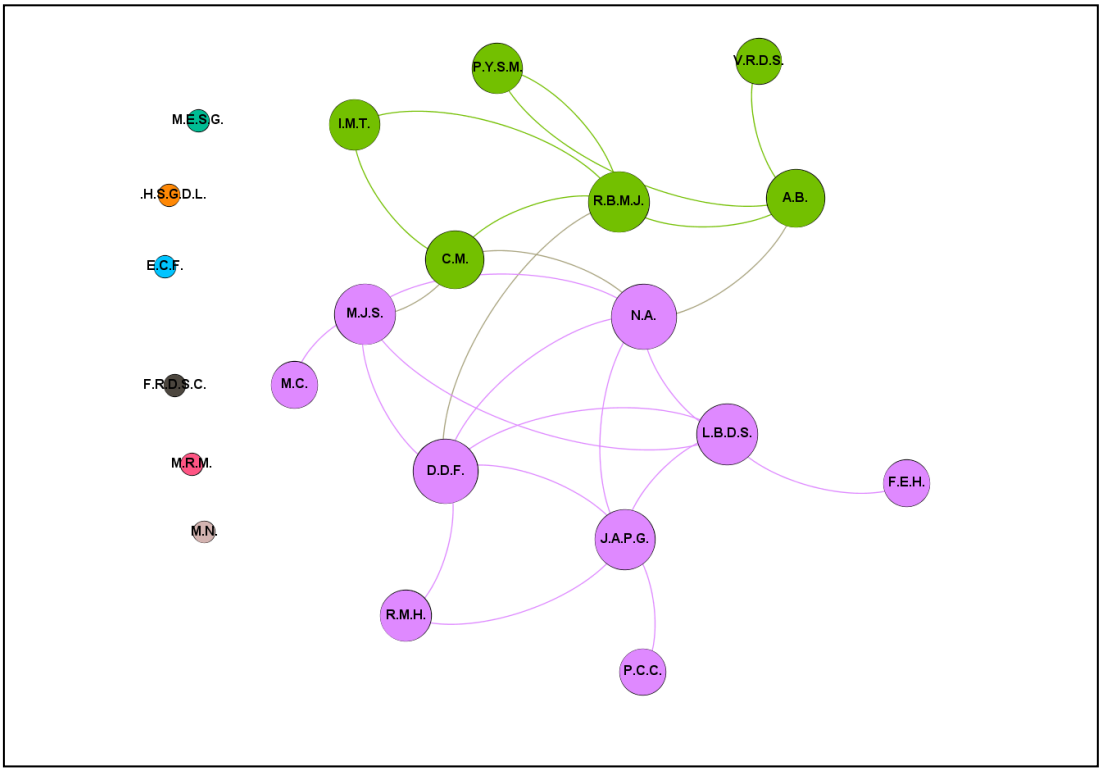


Oftalmologia Ciências Visuais (2010-2018)

Tecnologia Gestão e Saúde Ocular



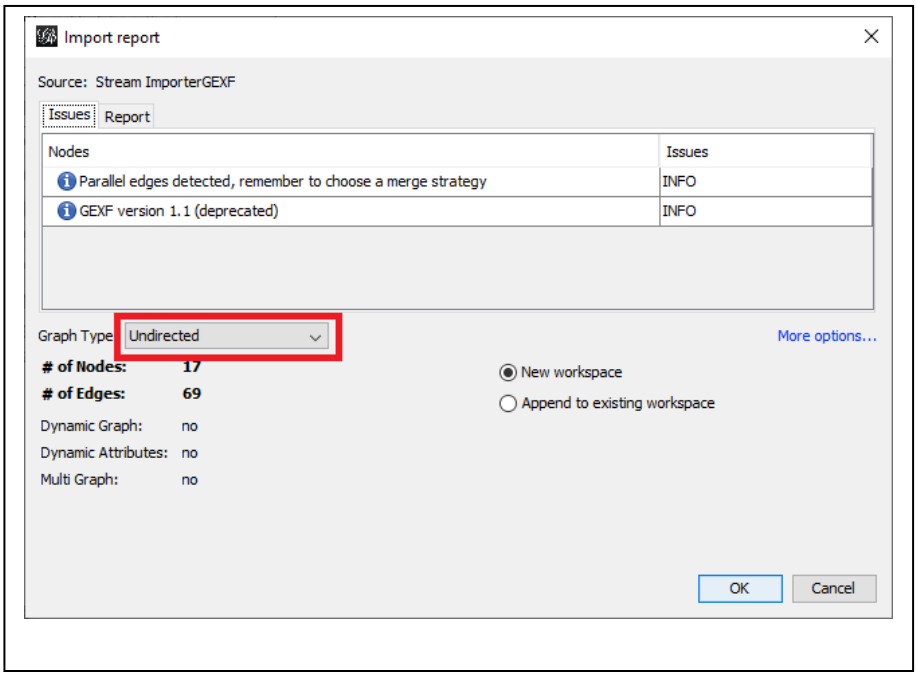
Tecnologia Gestão e Saúde Ocular (2010-2012)



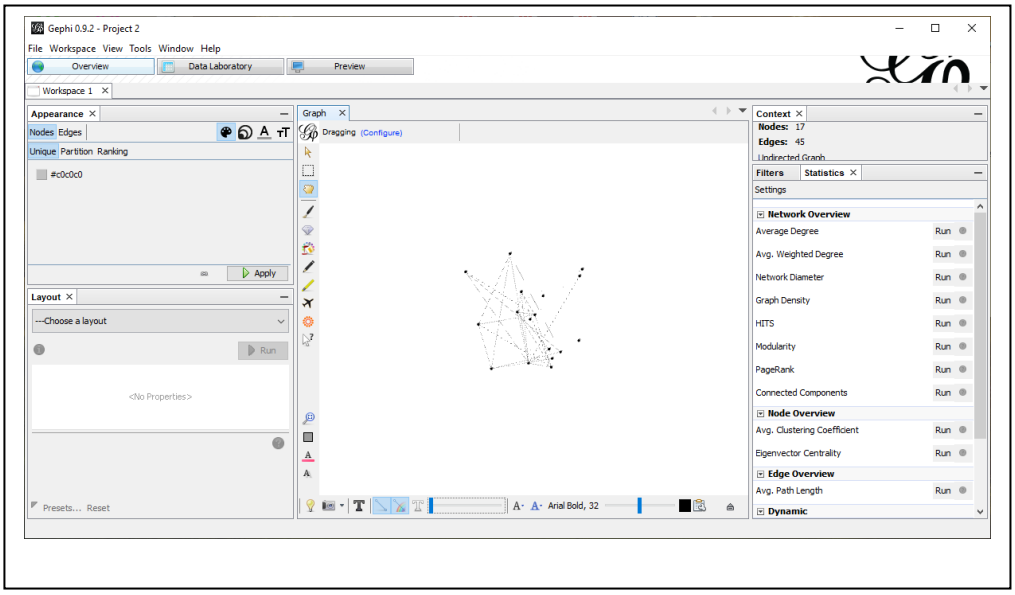
Tecnologia Gestão e Saúde Ocular (2010-2018)

Anexo 4 | Aplicação da rotina no Gephi

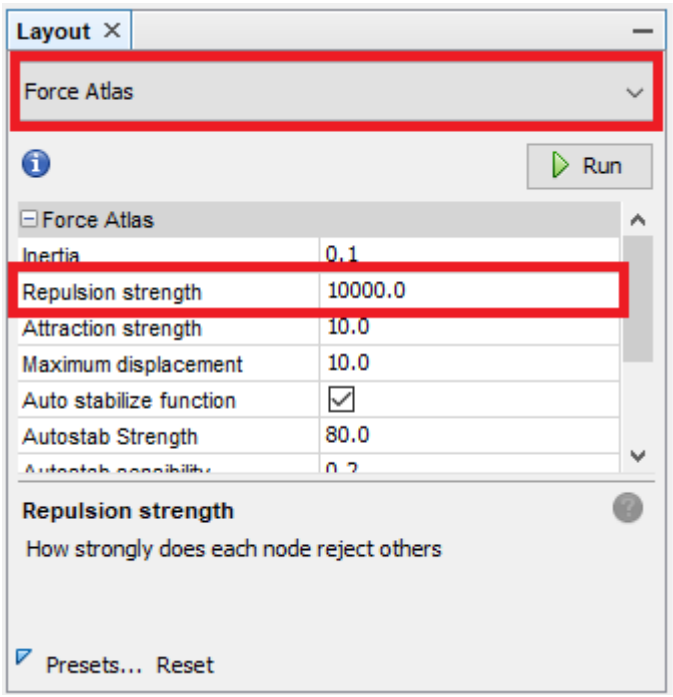
1 - Abrir arquivo no formato .gefx como grafo não direcionado



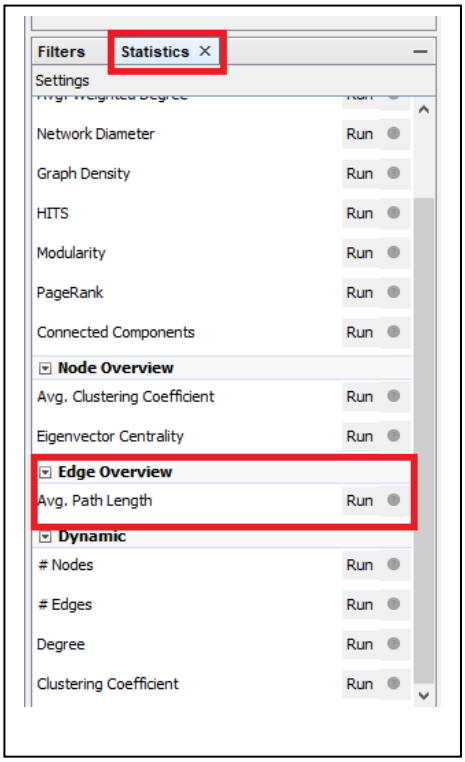
Será apresentado uma tela abaixo como abaixo com o dados relacionados conectados em vértices e arestas.



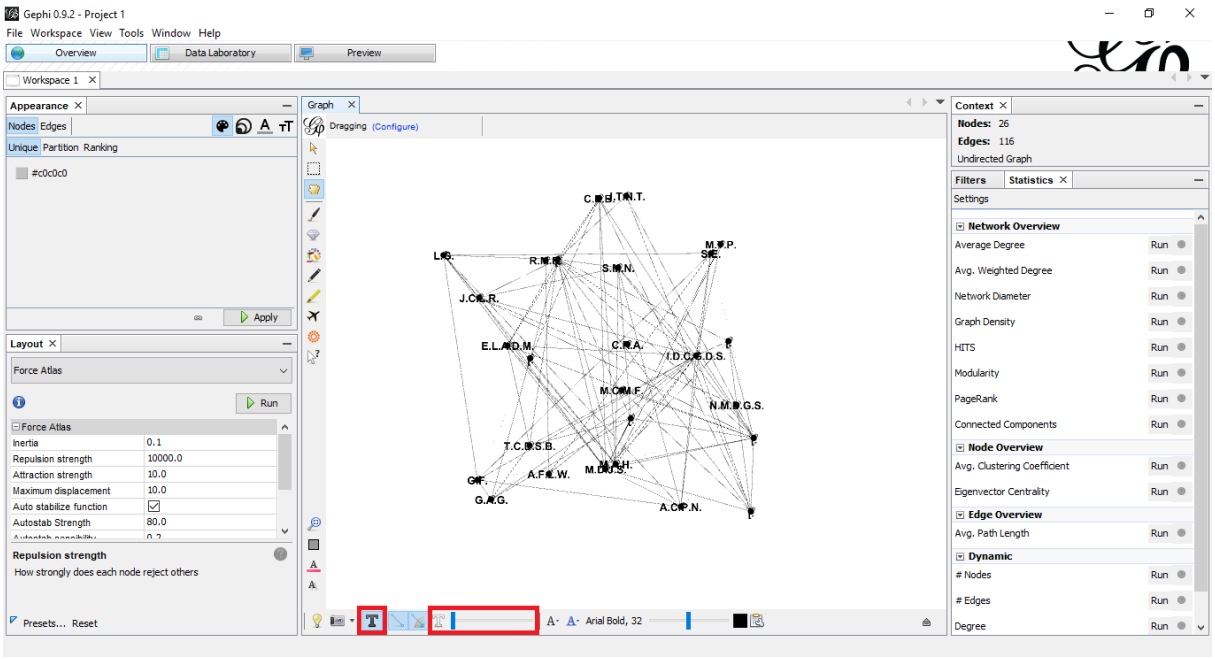
Localize a janela Layout no lado esquerdo da tela, selecione o algoritmo “Force Atlas” e altere a propriedade “Repulsion strength” para 10.000.



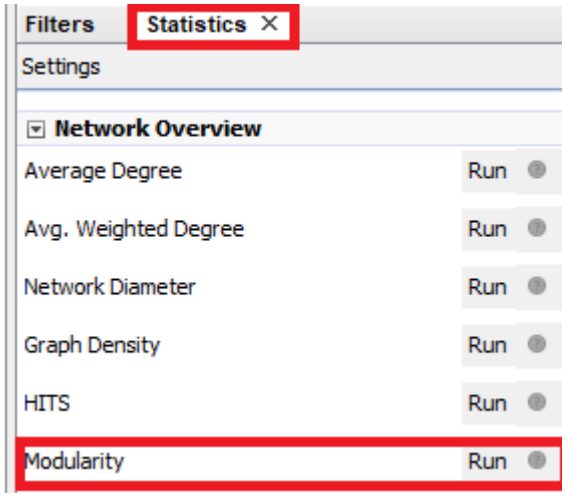
Localize a janela “Statistics” no lado direito e clique no botão “Run” no algoritmo “Avg. Path length”.



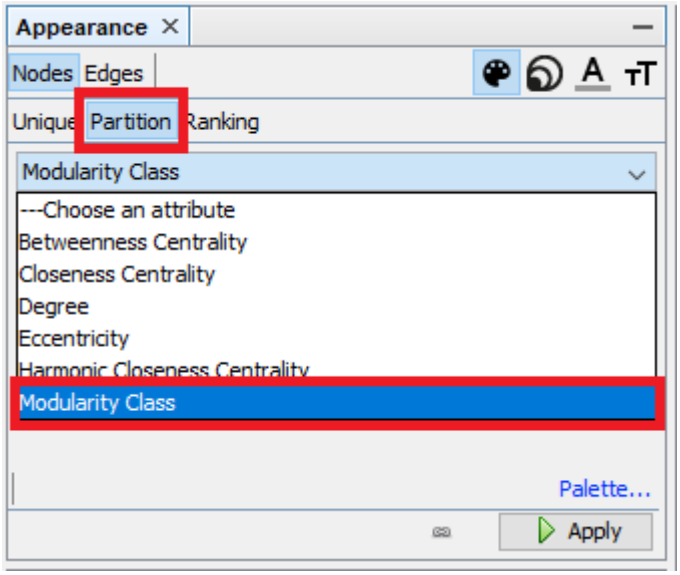
Ajuste valor do “Edge weigth scale” arrastando o “slide” do campo para deixar a grossura da aresta mais fina possível. Clique em “Show lable nodes” para que apareçam os nomes em cima dos vértices.



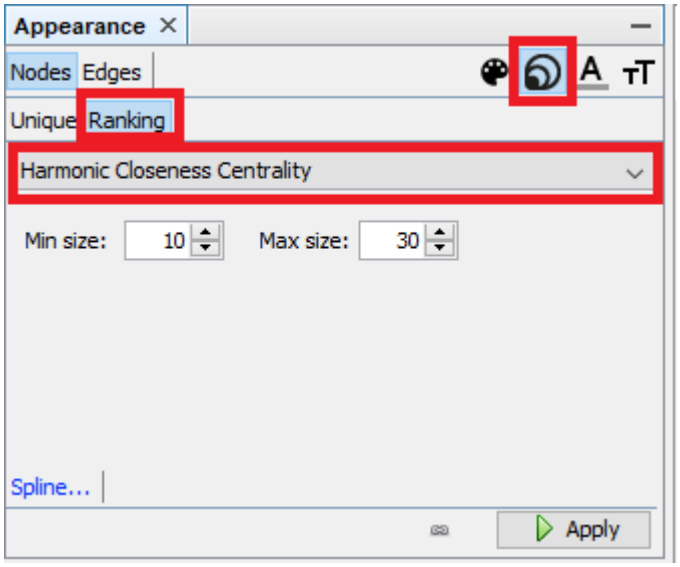
Execute o algoritmo “Modularity”. Clique no botão “Run”.



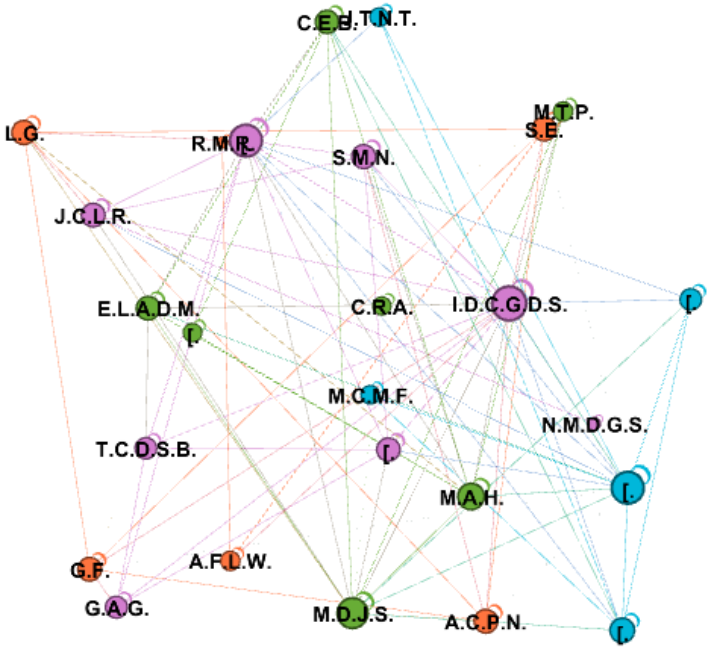
No lado esquerdo, na janela “Appearance”, selecione “Nodes”, “Partition”. Na lista de opções selecione “Modularity Class” e clique em “Apply”. Isso irá colorir o grafo com um algoritmo que relaciona cada cor com um grupo, baseado em ligações e semelhanças.



Para aumentar o tamanho dos vértices, baseado em interrelações com outros elementos, localize a tela “Appearance”, clique no símbolo de tamanho “Size”, selecione a aba “Ranking” e o algoritmo “Harmonic Closeness Centrality”. Clique em “Apply”.



Ao final deverá ter um resultado parecido com o abaixo:



Anexo 5 | Tabela de Grau de Relacionamento

Grau de relacionamento								
Pesquisador		Grau de relacionamento	G(0,0)	G(1,1)	G(2,5)	G(6,10)	G(11, 15)	G(16,20)
1	L.M.F.	20	0	0	0	0	0	1
2	M.D.J.S.	19	0	0	0	0	0	1
3	M.E.F.N.	17	0	0	0	0	0	1
4	M.G.F.S.	16	0	0	0	0	0	1
5	R.B.M.J.	16	0	0	0	0	0	1
6	A.L.H.D.L.F.	16	0	0	0	0	0	1
7	A.F.M.	15	0	0	0	1	0	0
8	P.S.	15	0	0	0	1	0	0
9	I.D.C.G.D.S.	14	0	0	0	1	0	0
10	E.A.J.	14	0	0	0	1	0	0
11	M.S.D.Q.C.	14	0	0	0	1	0	0
12	D.D.F.	14	0	0	0	1	0	0
13	R.M.	13	0	0	0	1	0	0
14	N.A.	13	0	0	0	1	0	0
15	M.J.B.C.G.	12	0	0	0	1	0	0
16	M.C.	12	0	0	0	1	0	0
17	M.R.T.	11	0	0	0	1	0	0
18	J.M.F.S.	11	0	0	0	1	0	0
19	M.M.	11	0	0	0	1	0	0
20	A.A.S.N.	10	0	0	0	0	1	0
21	D.F.V.	10	0	0	0	0	1	0
22	E.M.K.	10	0	0	0	0	1	0
23	R.P.B.	10	0	0	0	0	1	0
24	N.S.	10	0	0	0	0	1	0
25	S.R.S.	10	0	0	0	0	1	0
26	C.M.	10	0	0	0	0	1	0
27	F.F.	9	0	0	0	0	1	0
28	A.B.	9	0	0	0	0	1	0
29	L.B.D.S.	9	0	0	0	0	1	0
30	E.B.R.	9	0	0	0	0	1	0
31	W.N.	9	0	0	0	0	1	0
32	A.G.	8	0	0	0	0	1	0
33	D.D.C.A.	8	0	0	0	0	1	0
34	M.A.	8	0	0	0	0	1	0
35	R.D.A.C.	8	0	0	0	0	1	0
36	M.U.N.	8	0	0	0	0	1	0
37	E.D.S.	8	0	0	0	0	1	0
38	S.D.	8	0	0	0	0	1	0
39	L.M.M.N.	8	0	0	0	0	1	0

40	J.A.P.G.	8	0	0	0	0	1	0
41	A.P.J.	8	0	0	0	0	1	0
42	M.M.L.	7	0	0	0	0	1	0
43	J.L.M.	7	0	0	0	0	1	0
44	D.J.F.	7	0	0	0	0	1	0
45	R.J.A.	7	0	0	0	0	1	0
46	A.G.F.	7	0	0	0	0	1	0
47	E.T.O.	7	0	0	0	0	1	0
48	P.A.D.A.M.	7	0	0	0	0	1	0
49	W.C.A.D.S.	7	0	0	0	0	1	0
50	C.V.S.R.	7	0	0	0	0	1	0
51	M.J.S.	7	0	0	0	0	1	0
52	M.D.P.	6	0	0	0	0	1	0
53	M.S.N.	6	0	0	0	0	1	0
54	H.F.D.C.G.	6	0	0	0	0	1	0
55	E.S.	6	0	0	0	0	1	0
56	A.C.P.N.	6	0	0	0	0	1	0
57	E.L.A.D.M.	6	0	0	0	0	1	0
58	L.C.R.	6	0	0	0	0	1	0
59	L.C.G.	6	0	0	0	0	1	0
60	O.L.M.C.	6	0	0	0	0	1	0
61	E.G.L.T.	6	0	0	0	0	1	0
62	A.M.G.	5	0	0	1	0	0	0
63	M.O.T.	5	0	0	1	0	0	0
64	B.E.	5	0	0	1	0	0	0
65	Z.I.K.D.J.	5	0	0	1	0	0	0
66	S.Y.S.	5	0	0	1	0	0	0
67	F.F.G.	5	0	0	1	0	0	0
68	J.R.G.T.	5	0	0	1	0	0	0
69	A.P.C.	5	0	0	1	0	0	0
70	P.Y.S.M.	5	0	0	1	0	0	0
71	F.E.H.	5	0	0	1	0	0	0
72	R.M.H.	5	0	0	1	0	0	0
73	I.M.T.	5	0	0	1	0	0	0
74	T.D.S.P.	5	0	0	1	0	0	0
75	G.D.J.L.F.	4	0	0	1	0	0	0
76	H.P.	4	0	0	1	0	0	0
77	J.C.B.	4	0	0	1	0	0	0
78	E.B.G.	4	0	0	1	0	0	0
79	L.B.	4	0	0	1	0	0	0
80	G.A.G.	4	0	0	1	0	0	0
81	O.C.	4	0	0	1	0	0	0
82	J.E.J.	4	0	0	1	0	0	0
83	D.B.D.S.P.	4	0	0	1	0	0	0

84	R.M.Y.N.	4	0	0	1	0	0	0
85	N.D.O.P.	4	0	0	1	0	0	0
86	D.M.S.	4	0	0	1	0	0	0
87	P.I.L.	4	0	0	1	0	0	0
88	S.S.S.	3	0	0	1	0	0	0
89	I.H.J.K.	3	0	0	1	0	0	0
90	A.D.C.P.	3	0	0	1	0	0	0
91	P.S.B.	3	0	0	1	0	0	0
92	M.J.S.T.	3	0	0	1	0	0	0
93	A.C.A.	3	0	0	1	0	0	0
94	F.X.N.	3	0	0	1	0	0	0
95	R.S.D.O.F.	3	0	0	1	0	0	0
96	G.F.	3	0	0	1	0	0	0
97	J.T.N.T.	3	0	0	1	0	0	0
98	N.G.D.B.	3	0	0	1	0	0	0
99	S.T.Z.	3	0	0	1	0	0	0
100	C.A.F.G.	3	0	0	1	0	0	0
101	S.S.N.P.	3	0	0	1	0	0	0
102	F.L.M.H.	3	0	0	1	0	0	0
103	R.F.	3	0	0	1	0	0	0
104	V.R.D.S.	3	0	0	1	0	0	0
105	L.A.D.O.	3	0	0	1	0	0	0
106	M.N.N.B.J.	3	0	0	1	0	0	0
107	C.V.A.	2	0	0	1	0	0	0
108	L.A.R.	2	0	0	1	0	0	0
109	G.G.A.	2	0	0	1	0	0	0
110	M.W.	2	0	0	1	0	0	0
111	A.H.	2	0	0	1	0	0	0
112	D.N.	2	0	0	1	0	0	0
113	E.K.H.	2	0	0	1	0	0	0
114	C.R.A.	2	0	0	1	0	0	0
115	M.C.M.F.	2	0	0	1	0	0	0
116	M.B.P.	2	0	0	1	0	0	0
117	P.C.C.	2	0	0	1	0	0	0
118	F.R.D.S.C.	2	0	0	1	0	0	0
119	F.A.M.H.F.	1	0	1	0	0	0	0
120	F.M.J.	1	0	1	0	0	0	0
121	R.A.S.M.	1	0	1	0	0	0	0
122	J.W.	1	0	1	0	0	0	0
123	C.H.F.	1	0	1	0	0	0	0
124	E.N.D.C.C.	1	0	1	0	0	0	0
125	S.E.D.	1	0	1	0	0	0	0
126	I.D.S.	1	0	1	0	0	0	0
127	R.R.F.	1	0	1	0	0	0	0

128	M.C.J.	1	0	1	0	0	0	0
129	D.S.Z.	1	0	1	0	0	0	0
130	M.N.	1	0	1	0	0	0	0
131	M.C.	1	0	1	0	0	0	0
132	L.H.S.G.D.L.	1	0	1	0	0	0	0
133	M.C.	1	0	1	0	0	0	0
134	S.D.C.V.A.	0	1	0	0	0	0	0
135	C.A.R.	0	1	0	0	0	0	0
136	R.K.S.	0	1	0	0	0	0	0
137	I.L.F.	0	1	0	0	0	0	0
138	M.W.V.	0	1	0	0	0	0	0
139	N.M.D.G.S.	0	1	0	0	0	0	0
140	R.M.R.	0	1	0	0	0	0	0
141	R.P.	0	1	0	0	0	0	0
142	F.G.D.A.	0	1	0	0	0	0	0
143	H.F.Z.D.A.	0	1	0	0	0	0	0
144	W.F.A.	0	1	0	0	0	0	0
145	R.A.J.	0	1	0	0	0	0	0
146	E.C.F.	0	1	0	0	0	0	0
147	M.R.M.	0	1	0	0	0	0	0
148	M.E.S.G.	0	1	0	0	0	0	0
149	M.N.	0	1	0	0	0	0	0
			16	**15**	**57**	**13**	**42**	**6**

Anexo 6 | Resultado do Comitê de Ética de Pesquisa – UNIFESP

UNIFESP
UNIVERSIDADE FEDERAL DE SÃO PAULO

COMITÊ DE ÉTICA EM PESQUISA

SP

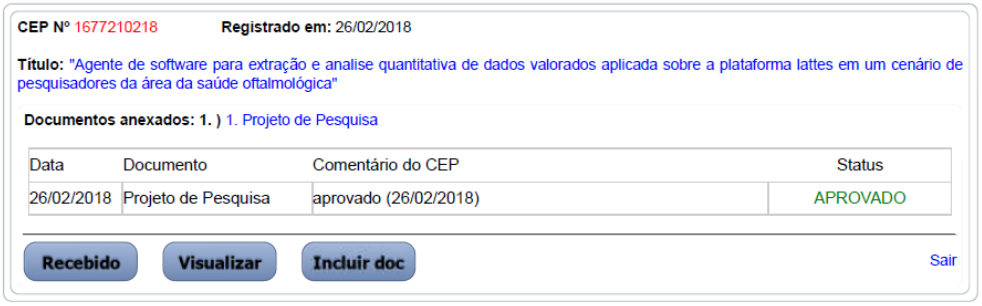
Estudo não envolvendo seres humanos ou animais

CPF:	11207931810	Característica:	Retrospectivo/Prospectivo
Título do projeto:	Agente de software para extração e analise quantitativa de dados valorados aplicada sobre a plataforma lattes em um cenário de pesquisadores da área da saúde oftalmológica		
Pesquisador:	Richard William Valdivia		
Celular:	11 967972889	e-mail:	rwvaldivia@yahoo.com.br
CV. Lattes:	http://lattes.cnpq.br/5686723676379001		
Depto/Disc:	Oftalmologia e Ciências Visuais	Campus:	Vila Clementino
Vínculo:	Aluno de pós-graduação		
Obj. Acadêmico:	Mestrado	Patente:	Não
Grande área:	Medicina	específica:	Oftalmologia
Patrocínio:	Ausente	Patrocinador:	
Orientador:	Profa. Dra. Maria Elisabete Salvador Graziosi	e-mail:	betesalvador@gmail.com
Chefe de Depto:	Prof. Dr. José Álvaro Pereira Gomes	e-mail:	japgomes13@gmail.com

Orçamento Financeiro

Descrição do item	Quantidade	Valor unitário (R$)
1. material de escritório	10	25,00
2. reprografia	300	7,00
3. material bibliográfico	5	80,00
4. correção de português	1	300,00
	Total	R$ 3.050,00

ARQUIVOS



CEP Nº 1677210218 **Registrado em:** 26/02/2018

Título: "Agente de software para extração e analise quantitativa de dados valorados aplicada sobre a plataforma lattes em um cenário de pesquisadores da área da saúde oftalmológica"

Documentos anexados: 1.) 1. Projeto de Pesquisa

Data	Documento	Comentário do CEP	Status
26/02/2018	Projeto de Pesquisa	aprovado (26/02/2018)	APROVADO

Recebido Visualizar Incluir doc Sair